

Argo data management
23 October 2003
ar-um-02-01

USER'S MANUAL

Version 1.0e

**WARNING : THIS
DOCUMENT IS A DRAFT
FOR A NEXT VERSION OF
THE USER'S MANUAL.**



part of the integrated global observation strategy



Argo data management

User's manual

Ref : ar-um-02-01

Ref ifremer : cor-do/dti-mut/02-084

Version : 1.0e

Date : 23/10/2003

Authors : Thierry Carval / Ifremer, Bob Keeley / MEDS, Yasushi Takatsuki / JAMSTEC, Takashi Yoshida / JMA, Stephen Loch / BODC, Claudia Schmid / AOML, Roger Goldsmith / WHOI, Annie Wong / PMEL, Rebecca McCreadie / BODC, Ann Thresher / CSIRO, Ahn Tran / MEDS

Table of contents

<u>HISTORY</u>	5
-----------------------	----------

<u>1. INTRODUCTION</u>	6
-------------------------------	----------

1.1. ARGO PROGRAM, DATA MANAGEMENT CONTEXT	6
1.2. ARGO FLOAT DATA	6
1.3. REAL-TIME AND DELAYED MODE DATA	8

<u>2. FORMATS DESCRIPTION</u>	9
--------------------------------------	----------

2.1. OVERVIEW OF THE FORMATS	9
-------------------------------------	----------

2.2. ARGO PROFILE FILE FORMAT 2.1	9
--	----------

2.2.1. DIMENSIONS AND DEFINITIONS	9
2.2.2. GENERAL INFORMATION ON THE PROFILE FILE	11
2.2.3. GENERAL INFORMATION FOR EACH PROFILE	11
2.2.4. MEASUREMENTS FOR EACH PROFILE	14
2.2.5. CALIBRATION INFORMATION FOR EACH PROFILE	16
2.2.6. HISTORY INFORMATION FOR EACH PROFILE	17

2.3. TRAJECTORY FORMAT 2.1	20
-----------------------------------	-----------

2.3.1. DIMENSIONS AND DEFINITIONS	20
2.3.2. GENERAL INFORMATION ON THE TRAJECTORY FILE	21
2.3.3. GENERAL INFORMATION ON THE FLOAT	21
2.3.4. LOCATIONS AND MEASUREMENTS FROM THE FLOAT	23
2.3.5. CYCLE INFORMATION FROM THE FLOAT	25
2.3.6. HISTORY INFORMATION	26

2.4. META-DATA FORMAT 2.1	28
----------------------------------	-----------

2.4.1. DIMENSIONS AND DEFINITIONS	28
2.4.2. GENERAL INFORMATION ON THE META-DATA FILE	29
2.4.3. FLOAT CHARACTERISTICS	30
2.4.4. FLOAT DEPLOYMENT AND MISSION INFORMATION	32
2.4.5. FLOAT SENSOR INFORMATION	33
2.4.6. FLOAT CALIBRATION INFORMATION	33
2.4.7. FLOAT CYCLE INFORMATION	35

2.5. TECHNICAL INFORMATION FORMAT 2.1	36
--	-----------

2.5.1. DIMENSIONS AND DEFINITIONS	36
2.5.2. GENERAL INFORMATION ON THE TECHNICAL DATA FILE	37
2.5.3. TECHNICAL DATA	38

2.6. GDAC FTP DIRECTORY FILE FORMAT 1.0	39
--	-----------

2.6.1. PROFILE DIRECTORY FORMAT	39
2.6.2. TRAJECTORY DIRECTORY FORMAT	40
2.6.3. META-DATA DIRECTORY FORMAT	41

<u>3. REFERENCE TABLES</u>	43
-----------------------------------	-----------

3.1. REFERENCE TABLE 1 : DATA TYPE	43
---	-----------

3.2. REFERENCE TABLE 2 : QUALITY CONTROL FLAGSCALE	43
---	-----------

3.3. REFERENCE TABLE 3 : PARAMETER CODE TABLE	44
3.4. REFERENCE TABLE 4 : DATA CENTRES AND INSTITUTIONS CODES	44
3.5. REFERENCE TABLE 5 : LOCATION CLASSES (ARGOS)	44
3.6. REFERENCE TABLE 6 : DATA STATE INDICATORS	45
3.7. REFERENCE TABLE 7 : HISTORY ACTION CODES	46
3.8. REFERENCE TABLE 8 : INSTRUMENT TYPES	47
3.9. REFERENCE TABLE 9 : POSITIONING SYSTEM	47
3.10. REFERENCE TABLE 10 : TRANSMISSION SYSTEM	47
3.11. REFERENCE TABLE 11 : QC TEST IDS	48
3.12. REFERENCE TABLE 12 : HISTORY SOFTWARE CODES	48
3.13. REFERENCE TABLE 13 : OCEAN CODES	49
<u>4. DATA ACCESS</u>	<u>50</u>
4.1. FILE NAMING CONVENTION ON GDACS	50
4.2. OTHER DATA SOURCES	50

History

Version	Date	Comment
0.9	29/12/2001	Thierry Carval : creation of the document
0.9a	18/01/2001	Bob Keeley : general comments and updates
0.9a	24/01/2002	Valérie Harscoat : general comments and updates
0.9a	25/01/2002	Claudia Schmid : general comments and updates
0.9a	24/01/2002	Roger Goldsmith : general comments and updates
0.9b	05/03/2002	Roger Goldsmith, Yasushi Takatsuki and Claudia Schmid comments implemented.
0.9c	24/04/2002	Comments from version 0.9b are implemented
1.0	09/07/2002	Comments from version 0.9c are implemented
1.0a	31/12/2002	Missing values in trajectory and calibration
1.0a	17/01/2003	Description of directory file format
1.0a	24/01/2003	Update of reference tables
1.0a	24/01/2003	Update of "measurements of each profile" to handle corrected values
1.0a	24/01/2003	Increase the size of DC_REFERENCE from STRING16 to STRING32
1.0b	17/03/2003	Replace corrected values with adjusted values
1.0b	29/04/2003	DC_REFERENCE removed from trajectory format general information of the float section
1.0b	30/04/2003	Use blank fill values for character variables
1.0c	30/04/2003	Proposal submitted on 30/04/2003
1.0d	14/08/2003	Proposal submitted on 14/08/2003 (green font)

1. Introduction

This document is the Argo data user's manual.

It contains the description of the formats and files produced by the data management group.

1.1. Argo program, data management context

Within a few years (2006), the Argo program will operate and manage a set of 3000 floats distributed in all oceans, with the vision that the network will be a permanent and operational system.

The Argo data management group is creating a unique data format for internet distribution to users and for data exchange between national data centres (DACS) and global data centres (GDACs).

Profile data, metadata, trajectories and technical data are included in this standardization effort.

The Argo data formats are based on NetCDF because :

- It is a widely accepted data format by the user community,
- It is a self-describing format for which tools are widely available,
- It is a reliable and efficient format for data exchange.

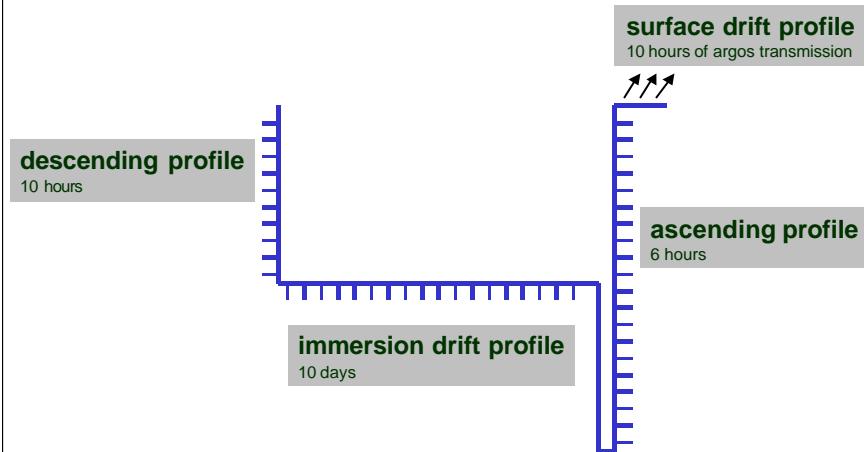
1.2. Argo float data

An Argo float drifts for a number of years in the ocean. It continuously performs measurement cycles. Each cycle lasts about 10 days and can be divided into 4 phases :

- A descent from surface to a defined pressure (eg : 1500 decibars),
- A subsurface drift (eg : 10 days),
- An ascending profile with measurements (eg : pressure, temperature, salinity),
- A surface drift with data transmission to a communication satellite.

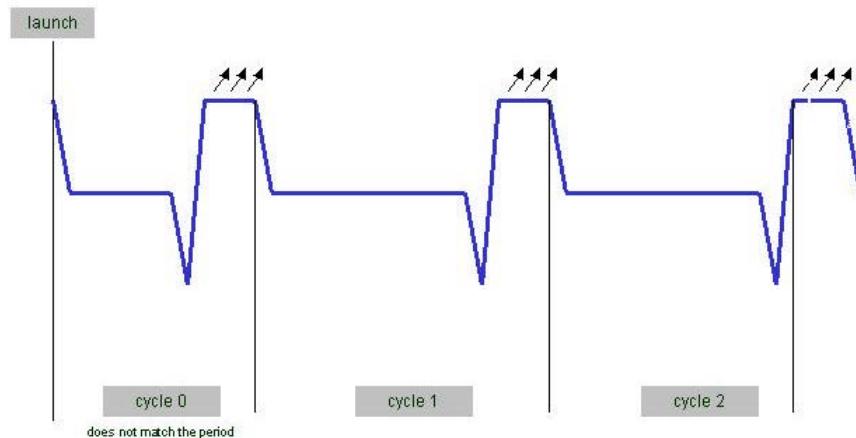
argo data management

An argo profiler cycle



argo data management

Argo profiler cycling



1.3. Real-time and Delayed mode data

Data from Argo floats are transmitted from the float, passed through processing and automatic quality control procedures as quickly as possible after the float begins reporting at the surface. The target is to issue the data to the GTS and Global Data servers within 24 hours of surfacing, or as quickly thereafter as possible. These are called real-time data.

The data are also issued to the Principle Investigators on the same schedule as they are sent to the Global servers. These scientists apply other procedures to check data quality and the target is for these data to be returned to the global data centres within 6 months. These constitute the delayed mode data.

2. Formats description

2.1. Overview of the formats

Argo data formats are based on NetCDF from Ucar.

NetCDF (network Common Data Form) is an interface for array-oriented data access and a library that provides an implementation of the interface. The NetCDF library also defines a machine-independent format for representing scientific data. Together, the interface, library, and format support the creation, access, and sharing of scientific data. The NetCDF software was developed at the Unidata Program Centre in Boulder, Colorado. The [freely available](#) source can be obtained as [a compressed tar file](#) or [a zip file](#) from Unidata or from other [mirror sites](#).

- Ucar web site address : <http://www.ucar.edu/ucar>
- NetCDF documentation :
<http://www.unidata.ucar.edu/packages/netcdf/index.html>

Argo formats are divided in 4 sections :

- Dimensions and definitions
- General information
- Data section
- History section

The Argo NetCDF formats do not contain any global attribute.

Argo date and times : all date and time have to be explained in UTC time, universal time coordinates.

2.2. Argo profile file format 2.1

An Argo profile file contains a set of profiles. The minimum number is one profile. There is no defined maximum number of profiles.

A profile contains measurements performed at different pressures by an Argo float.

A profile contains typically 100 pressures, from 0 decibar (surface) to 2000 decibars (approximately 2000 meters depth).

For each pressure sample, there is a fixed number of parameters measured or calculated such as temperature, salinity or conductivity.

For file naming conventions, see §4.1 .

2.2.1. Dimensions and definitions

Name	Value	Definition
DATE_TIME	DATE_TIME = 14;	This dimension is the length of an ASCII date and time value. Date_time convention is : YYYYMMDDHHMISS • YYYY : year

		<ul style="list-style-type: none"> • MM : month • DD : day • HH : hour of the day (as 0 to 23) • MI : minutes (as 0 to 59) • SS : seconds (as 0 to 59) <p>Date and time values are always in universal time coordinates (UTC). Examples : 20010105172834 : January 5th 2001 17:28:34 19971217000000 : December 17th 1997 00:00:00</p>
STRING256 STRING64 STRING32 STRING16 STRING8 STRING4 STRING2	STRING256 = 256; STRING64 = 64; STRING32 = 32; STRING16 = 16; STRING8 = 8; STRING4 = 4; STRING2 = 2;	String dimensions from 2 to 256.
N_PROF	N_PROF = <int value>;	<p>Number of profiles contained in the file. This dimension depends on the data set. A file contains at least one profile. There is no defined limit on the maximum number of profiles in a file. Example : N_PROF = 100</p>
N_PARAM	N_PARAM = <int value>;	<p>Maximum number of parameters measured or calculated for a pressure sample. This dimension depends on the data set. Examples : (pressure, temperature) : N_PARAM = 2 (pressure, temperature, salinity) : N_PARAM = 3 (pressure, temperature, conductivity, salinity) : N_PARAM = 4</p>
N_LEVELS	N_LEVELS = <int value>;	<p>Maximum number of pressure levels contained in a profile. This dimension depends on the data set. Example : N_LEVELS = 100</p>
N_CALIB	N_CALIB = <int value>;	<p>Maximum number of calibrations performed on a profile. This dimension depends on the data set. Example : N_CALIB = 10</p>
N_HISTORY	N_HISTORY = UNLIMITED;	Number of history records.

2.2.2. General information on the profile file

This section contains information about the whole file.

Name	Definition	Comment
DATA_TYPE	char DATA_TYPE(STRING16); DATA_TYPE:comment = "Data type"; DATA_TYPE:_FillValue = " ";	This field contains the type of data contained in the file. The list of acceptable data types is in the reference table 1. Example : Argo profile
FORMAT_VERSION	char FORMAT_VERSION(STRING4); FORMAT_VERSION:comment = "File format version "; FORMAT_VERSION:_FillValue = " ";	File format version Example : «2.1»
HANDBOOK_VERSION	char HANDBOOK_VERSION(STRING4); HANDBOOK_VERSION:comment = "Data handbook version"; HANDBOOK_VERSION:_FillValue = " ";	Version number of the data handbook. This field indicates that the data contained in this file are managed according to the policy described in the Argo data management handbook. Example : «1.0»
REFERENCE_DATE_TIME	char REFERENCE_DATE_TIME(DATE_TIME); REFERENCE_DATE_TIME:comment = "Date of reference for Julian days"; REFERENCE_DATE_TIME:conventions = "YYYYMMDDHHMISS"; REFERENCE_DATE_TIME:_FillValue = " ";	Date of reference for julian days. The recommended reference date time is "19500101000000" : January 1 st 1950 00:00:00

2.2.3. General information for each profile

This section contains general information on each profile.

Each item of this section has a N_PROF (number of profiles) dimension.

Name	Definition	Comment
PLATFORM_NUMBER	char PLATFORM_NUMBER(N_PROF, STRING8); PLATFORM_NUMBER:long_name = "Float unique identifier"; PLATFORM_NUMBER:conventions = "WMO float identifier : A9IIIII"; PLATFORM_NUMBER:_FillValue = " ";	WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique. Example : 6900045
PROJECT_NAME	char PROJECT_NAME(N_PROF, STRING64); PROJECT_NAME:comment = "Name of the project"; PROJECT_NAME:_FillValue = " ";	Name of the project which operates the profiling float that performed the profile. Example : GYROSCOPE (EU project for ARGO program)
PI_NAME	char PI_NAME (N_PROF, STRING64); PI_NAME:comment = "Name of the principal investigator"; PI_NAME:_FillValue = " ";	Name of the principal investigator in charge of the profiling float. Example : Yves Desaubies
STATION_PARAMETERS	char STATION_PARAMETERS(N_PROF, N_PARAM, STRING4); STATION_PARAMETERS:long_name = "List of available parameters for the station"; STATION_PARAMETERS:conventions = "Argo reference table 3"; STATION_PARAMETERS:_FillValue = " ";	List of parameters contained in this profile. The parameter names are listed in reference table 3. Examples : TEMP, PSAL, CNDC TEMP : temperature PSAL : practical salinity CNDC : conductivity
CYCLE_NUMBER	int CYCLE_NUMBER(N_PROF); CYCLE_NUMBER:long_name = "Float cycle number"; CYCLE_NUMBER:conventions = "0..N, 0 : launch cycle (if exists), 1 : first complete cycle"; CYCLE_NUMBER:_FillValue = 99999;	Float cycle number. A profiling float performs cycles. In each cycle, it performs an ascending vertical profile, a subsurface drift and a surface drift. In some cases, it also performs a descending vertical profile. 0 is the number of the launch cycle. The

		subsurface drift of the cycle 0 may not be complete. 1 is the number of the first complete cycle. Example : 10 : cycle number 10
DIRECTION	char DIRECTION(N_PROF); DIRECTION:long_name = "Direction of the station profiles"; DIRECTION:conventions = "A: ascending profiles, D: descending profiles "; DIRECTION:_FillValue = " ";	Type of profile on which measurement occurs. A : ascending profile D : descending profile
DATA_CENTRE	char DATA_CENTRE(N_PROF, STRING2); DATA_CENTRE:long_name = "Data centre in charge of float data processing"; DATA_CENTRE:conventions = "Argo reference table 4"; DATA_CENTRE:_FillValue = " ";	Code for the data centre in charge of the float data management. The data centre codes are described in the reference table 4. Example : ME for MEDS
DATE_CREATION	char DATE_CREATION(DATE_TIME); DATE_CREATION:comment = "Date of file creation "; DATE_CREATION:conventions = "YYYYMMDDHHMISS"; DATE_CREATION:_FillValue = " ";	Date and time (UTC) of creation of this file. Format : YYYYMMDDHHMISS Example : 20011229161700 : December 29 th 2001 16 :17 :00
DATE_UPDATE	char DATE_UPDATE(DATE_TIME); DATE_UPDATE:long_name = "Date of update of this file"; DATE_UPDATE:conventions = "YYYYMMDDHHMISS"; DATE_UPDATE:_FillValue = " ";	Date and time (UTC) of update of this file. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 09 :05 :00
DC_REFERENCE	char DC_REFERENCE(N_PROF, STRING32); DC_REFERENCE:long_name = "Station unique identifier in data centre"; DC_REFERENCE:conventions = "Data centre convention"; DC_REFERENCE:_FillValue = " ";	Unique identifier of the profile in the data centre. Data centres may have different identifier schemes. DC_REFERENCE is therefore not unique across data centres.
DATA_STATE_INDICATOR	char DATA_STATE_INDICATOR(N_PROF, STRING4); DATA_STATE_INDICATOR:long_name = "Degree of processing the data have passed through"; DATA_STATE_INDICATOR:conventions = "Argo reference table 6"; DATA_STATE_INDICATOR:_FillValue = " ";	Degree of processing the data has passed through. The data state indicator is described in the reference table 6.
DATA_MODE	char DATA_MODE(N_PROF); DATA_MODE:long_name = "Delayed mode or real time data"; DATA_MODE:conventions = "R : real time; D : delayed mode";	Indicates if the profile contains real time or delayed mode data. R : real time data D : delayed mode data
INST_REFERENCE	char INST_REFERENCE(N_PROF, STRING64); INST_REFERENCE:long_name = "Instrument type"; INST_REFERENCE:conventions = "Brand, type, serial number"; INST_REFERENCE:_FillValue = " ";	References of the instrument : brand, type, serial number Example : APEX-SBE 259
WMO_INST_TYPE	char WMO_INST_TYPE(N_PROF, STRING4); WMO_INST_TYPE:long_name = "Coded instrument type"; WMO_INST_TYPE:conventions = "Argo reference table 8"; WMO_INST_TYPE:_FillValue = " ";	Instrument type from WMO code table 1770. A subset of WMO table 1770 is documented in the reference table 8. Example : 846 : Webb Research float, Seabird sensor
JULD	double JULD(N_PROF); JULD:long_name = "Julian day (UTC) of the station relative to REFERENCE_DATE_TIME";	Julian day of the profile ¹ . The integer part represents the day, the decimal part represents the time of the profile.

¹ Assume that a float profiles on its ascent. When the float first comes to the surface, it begins to transmit data. Each data transmission has a time attached to it and the earliest time is what is recorded in JULD. It is possible that the first transmission from a float cannot be used to derive its location. In this case, the time of location, JULD_LOCATION, is different and later than the time of the profile.

	JULD:units = "days since 1950-01-01 00:00:00 UTC"; JULD:conventions = "Relative julian days with decimal part (as parts of day)"; JULD:_FillValue = 999999.;	Date and time are in universal time coordinates. The julian day is relative to REFERENCE_DATE_TIME. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_QC	char JULD_QC(N_PROF); JULD_QC:long_name = "Quality on Date and Time"; JULD_QC:conventions = "Argo reference table 2"; JULD_QC:_FillValue = "0";	Quality flag on JULD date and time. The flag scale is described in the reference table 2. Example : 1 : the date and time seems correct.
JULD_LOCATION	double JULD_LOCATION(N_PROF); JULD:long_name = "Julian day (UTC) of the location relative to REFERENCE_DATE_TIME "; JULD:units = " days since 1950-01-01 00:00:00 UTC"; JULD:conventions = "Relative julian days with decimal part (as parts of day)"; JULD:_FillValue = 999999.;	Julian day of the location of the profile. The location of the profile is generally estimated after the end of the profile. The date of the profile and the date of the location of the profile can be therefore slightly different. The integer part represents the day, the decimal part represents the time of the profile. Date and time are in universal time coordinates. The julian day is relative to REFERENCE_DATE_TIME. Example : 18833.8013889885 : July 25 2001 19:14:00
LATITUDE	double LATITUDE(N_PROF); LATITUDE:long_name = "Latitude of the station, best estimate"; LATITUDE:units = "degree_north"; LATITUDE:_FillValue = 99999.; LATITUDE:valid_min = -90.; LATITUDE:valid_max = 90.;	Latitude of the profile. Unit : degree north This field contains the best estimated latitude. The latitude value may be improved in delayed mode. The measured locations of the float are located in the trajectory file. Example : 44.4991 : 44° 29' 56.76" N
LONGITUDE	double LONGITUDE(N_PROF); LONGITUDE:long_name = "Longitude of the station, best estimate"; LONGITUDE:units = "degree_east"; LONGITUDE:_FillValue = 99999.; LONGITUDE:valid_min = -180.; LONGITUDE:valid_max = 180.;	Longitude of the profile. Unit : degree east This field contains the best estimated longitude. The longitude value may be improved in delayed mode. The measured locations of the float are located in the trajectory file. Example : 16.7222 : 16° 43' 19.92" E
POSITION_QC	char POSITION_QC(N_PROF); POSITION_QC:long_name = "Quality on position (latitude and longitude)"; POSITION_QC:conventions = "Argo reference table 2"; POSITION_QC:_FillValue = "0";	Quality flag on position. The flag on position is set according to (LATITUDE, LONGITUDE, JULD_LOCATION) quality. The flag scale is described in the reference table 2. Example : 1 : position seems correct.
POSITIONING_SYSTEM	char POSITIONING_SYSTEM(N_PROF, STRING8); POSITIONING_SYSTEM:long_name = "Positioning system"; POSITIONING_SYSTEM:_FillValue = " ";	Name of the system in charge of positioning the float locations from reference table 9. Examples : ARGOS
PROFILE_<PARAM>_QC	char PROFILE_<PARAM>_QC(N_PROF); PROFILE_<PARAM>_QC:long_name = "Global quality flag of <PARAM> profile"; PROFILE_<PARAM>_QC:conventions = "Argo reference table 2"; PROFILE_<PARAM>_QC:_FillValue = "0";	Global quality flag on the PARAM profile. PARAM is among the STATION_PARAMETERS. The flag scale is described in the reference table 2. The overall flag is set to be the worst flag found at any level in the profile. From best to worst the flag order is 1, 5, 2, 3, 4, 0. Example : PROFILE_TEMP_QC = 1 : the temperature profile seems correct. PROFILE_PSAL_QC = 1 : the salinity profile seems correct.

2.2.4. Measurements for each profile

This section contains information on each level of each profile.

Each variable in this section has a N_PROF (number of profiles), N_LEVELS (number of pressure levels) dimension.

The original data received from the float and examined by real-time quality control should be placed in the <PARAM> and the QC flags set by the real-time process should be placed in the <PARAM>_QC field. The values and flags in the <PARAM> fields should never be altered.

Each parameter can be adjusted in delayed-mode. In that case <PARAM> contains the original values, <PARAM>_ADJUSTED contains the adjusted values, <PARAM>_ADJUSTED_QC contains the QC flags set by the delayed-mode process, and <PARAM>_ADJUSTED_ERROR contains the adjustment uncertainties.

Name	Definition	Comment
<PARAM>	float <PARAM>(N_PROF, N_LEVELS); <PARAM>:long_name = "<X>"; <PARAM>:_FillValue = <X>; <PARAM>:units = "<X>"; <PARAM>:valid_min = <X>; <PARAM>:valid_max = <X>; <PARAM>:comment = "<X>"; <PARAM>:C_format = "<X>"; <PARAM>:FORTRAN_format = "<X>"; <PARAM>:resolution = <X>;	<PARAM> contains the original values of a parameter listed in reference table 3. <X> : this field is specified in the reference table 3.
<PARAM>_QC	char <PARAM>_QC(N_PROF, N_LEVELS); <PARAM>_QC:long_name = "quality flag"; <PARAM>_QC:conventions = "Argo reference table 2"; <PARAM>_QC:_FillValue = "0"	Quality flag applied on each <PARAM> values. The flag scale is specified in table 2.
<PARAM>_ADJUSTED	float <PARAM>_ADJUSTED(N_PROF, N_LEVELS); <PARAM>_ADJUSTED:long_name = "<X>"; <PARAM>_ADJUSTED:_FillValue = <X>; <PARAM>_ADJUSTED:units = "<X>"; <PARAM>_ADJUSTED:valid_min = <X>; <PARAM>_ADJUSTED:valid_max = <X>; <PARAM>_ADJUSTED:comment = "<X>"; <PARAM>_ADJUSTED:C_format = "<X>"; <PARAM>_ADJUSTED:FORTRAN_format = "<X>"; <PARAM>_ADJUSTED:resolution= <X>;	<PARAM>_ADJUSTED contains the adjusted values derived from the original values of the parameter. <X> : this field is specified in the reference table 3. <PARAM>_ADJUSTED is not mandatory.
<PARAM>_ADJUSTED_QC	char <PARAM>_ADJUSTED_QC(N_PROF, N_LEVELS); <PARAM>_ADJUSTED_QC:long_name = "quality flag"; <PARAM>_ADJUSTED_QC:conventions = "Argo reference table 2"; <PARAM>_ADJUSTED_QC:_FillValue = "0"	Quality flag applied on each <PARAM>_ADJUSTED values. The flag scale is specified in reference table 2. <PARAM>_ADJUSTED_QC is not mandatory.
<PARAM>_ADJUSTED_ERROR	float <PARAM>_ADJUSTED_ERROR(N_PROF, N_LEVELS); <PARAM>_ADJUSTED_ERROR:long_name = "<X>"; <PARAM>_ADJUSTED_ERROR:_FillValue = <X>; <PARAM>_ADJUSTED_ERROR:units = "<X>"; <PARAM>_ADJUSTED_ERROR:comment = "Contains the error on the adjusted values as determined by the scientific QC process. This field is not mandatory if scientific QC has not been applied."; <PARAM>_ADJUSTED_ERROR:C_format = "<X>"; <PARAM>_ADJUSTED_ERROR:FORTRAN_format = "<X>"; <PARAM>_ADJUSTED_ERROR:resolution= <X>;	<PARAM>_ADJUSTED_ERROR contains the error on the adjusted values of the parameter. <X> : this field is specified in the reference table 3. <PARAM>_ADJUSTED_ERROR is not mandatory.

Example of a profiling float performing temperature measurements with adjusted values of temperature :

Parameter definition : PRES, TEMP, TEMP_ADJUSTED

```

float PRES(N_PROF, N_LEVELS);
PRES:long_name = "SEA PRESSURE (sea surface = 0)";
PRES:_FillValue = 99999.f;
PRES:units = "decibar";
PRES:valid_min = 0.f;
PRES:valid_max = 1200.f;
PRES:comment = "In situ measurement, sea surface = 0";
PRES:C_format = "7.1f";
PRES:FORTRAN_format= "F7.1";
PRES:resolution= 0.1f;

char PRES_QC(N_PROF, N_LEVELS);
PRES_QC:long_name = "quality flag";
PRES_QC:conventions = "Argo reference table 2";
PRES_QC:_FillValue = "0";

float TEMP(N_PROF, N_LEVELS);
TEMP:long_name = "SEA TEMPERATURE";
TEMP:_FillValue = 99999.f;
TEMP:units = "degree_Celsius";
TEMP:valid_min = -2.f;
TEMP:valid_max = 40.f;
TEMP:comment = "In situ measurement";
TEMP:C_format = "%9.3f";
TEMP:FORTRAN_format = "F9.3";
TEMP:resolution = 0.001f;

char TEMP_QC(N_PROF, N_LEVELS);
TEMP_QC:long_name = "quality flag";
TEMP_QC:conventions = "Argo reference table 2";
TEMP_QC:_FillValue = "0";

float TEMP_ADJUSTED(N_PROF, N_LEVELS);
TEMP_ADJUSTED:long_name = "ADJUSTED SEA TEMPERATURE";
TEMP_ADJUSTED:_FillValue = 99999.f;
TEMP_ADJUSTED:units = "degree_Celsius";
TEMP_ADJUSTED:valid_min = -2.f;
TEMP_ADJUSTED:valid_max = 40.f;
TEMP_ADJUSTED:comment = "Adjusted parameter";
TEMP_ADJUSTED:C_format = "%9.3f";
TEMP_ADJUSTED:FORTRAN_format= "F9.3";
TEMP_ADJUSTED:resolution= 0.001f;

char TEMP_ADJUSTED_QC(N_PROF, N_LEVELS);
TEMP_ADJUSTED_QC:long_name = "quality flag";
TEMP_ADJUSTED_QC:conventions = "Argo reference table 2";
TEMP_ADJUSTED_QC:_FillValue = "0";

float TEMP_ADJUSTED_ERROR(N_PROF, N_LEVELS);
TEMP_ADJUSTED_ERROR:long_name = "ERROR ON ADJUSTED SEA TEMPERATURE";
TEMP_ADJUSTED_ERROR:_FillValue = 99999.f;
TEMP_ADJUSTED_ERROR:units = "degree_Celsius";
TEMP_ADJUSTED_ERROR:comment = " Contains the error on the adjusted values as determined by the scientific QC process. This field is not mandatory if scientific QC has not been applied.";
TEMP_ADJUSTED_ERROR:C_format = "%9.3f";
TEMP_ADJUSTED_ERROR:FORTRAN_format= "F9.3";
TEMP_ADJUSTED_ERROR:resolution= 0.001f;

```

2.2.5. Calibration information for each profile

Calibrations are applied to parameters to create adjusted parameters. Different calibration methods will be used by groups processing Argo data. When a method is applied, its description is stored in the following fields.

Example : in a temperature profile (TEMP), the calibrated values are stored in a adjusted profile (TEMP_ADJUSTED).

This section contains calibration information for each parameter of each profile.

Each item of this section has a N_PROF (number of profiles), N_CALIB (number of calibrations), N_PARAM (number of parameters) dimension.

Name	Definition	Comment
PARAMETER	char PARAMETER(N_PROF, N_CALIB, N_PARAM,STRING4); PARAMETER:long_name = "List of parameters with calibration information"; PARAMETER:conventions = "Argo reference table 3"; PARAMETER:_FillValue = " ";	Name of the calibrated parameter. The list of parameters is in reference table 3. Example : PSAL
SCIENTIFIC_CALIB_EQUATION	Char SCIENTIFIC_CALIB_EQUATION(N_PROF, N_CALIB, N_PARAM,STRING256); SCIENTIFIC_CALIB_EQUATION:long_name = "Calibration equation for this parameter"; SCIENTIFIC_CALIB_EQUATION:_FillValue = " ";	Calibration equation applied to the parameter. Example : $T_c = a_1 * T + a_0$
SCIENTIFIC_CALIB_COEFFICIENT	Char SCIENTIFIC_CALIB_COEFFICIENT(N_PROF, N_CALIB, N_PARAM,STRING256); SCIENTIFIC_CALIB_COEFFICIENT:long_name = "Calibration coefficients for this equation"; SCIENTIFIC_CALIB_COEFFICIENT:_FillValue = " ";	Calibration coefficients for this equation. Example : $a_1=0.99997$, $a_0=0.0021$
SCIENTIFIC_CALIB_COMMENT	Char SCIENTIFIC_CALIB_COMMENT(N_PROF, N_CALIB, N_PARAM,STRING256); SCIENTIFIC_CALIB_COMMENT:long_name = "Comment applying to this parameter calibration"; SCIENTIFIC_CALIB_COMMENT:_FillValue = " ";	Comment about this calibration Example : The sensor is not stable
CALIBRATION_DATE	Char CALIBRATION_DATE (N_PROF N_CALIB, N_PARAM, DATE_TIME) CALIBRATION_DATE:_FillValue = "0000000000000000";	Date of the calibration. Example : 20011217161700

When no calibration information is available :

- N_CALIB is set to 1
- SCIENTIFIC_CALIB_EQUATION, SCIENTIFIC_CALIB_COEFFICIENT, SCIENTIFIC_CALIB_COMMENT are set to “none”
- CALIBRATION_DATE is set to “0000000000000000”

2.2.6. History information for each profile

This section contains history information for each action performed on each profile by a data centre.

Each item of this section has a N_HISTORY (number of history records), N_PROF (number of profiles) dimension.

A history record is created whenever an action is performed on a profile.

The recorded actions are coded and described in the history code table from the reference table 7.

Name	Definition	Comment
HISTORY_INSTITUTION	char HISTORY_INSTITUTION (N_HISTORY, N_PROF, STRING4); HISTORY_INSTITUTION:long_name = "Institution which performed action"; HISTORY_INSTITUTION:conventions = "Argo reference table 4"; HISTORY_INSTITUTION:_FillValue = " ";	Institution that performed the action. Institution codes are described in reference table 4. Example : ME for MEDS
HISTORY SOFTWARE	char HISTORY SOFTWARE(N_HISTORY, N_PROF, STRING4); HISTORY SOFTWARE:long_name = "Software which performed action"; HISTORY SOFTWARE:conventions = "Institution dependent"; HISTORY SOFTWARE:_FillValue = " ";	Name of the software that performed the action. Example : QCA1
HISTORY SOFTWARE_RELEASE	char HISTORY SOFTWARE_RELEASE(N_HISTORY, N_PROF, STRING4); HISTORY SOFTWARE_RELEASE:long_name = "Version/release of software which performed action"; HISTORY SOFTWARE_RELEASE:conventions = "Institution dependent"; HISTORY SOFTWARE_RELEASE:_FillValue = " ";	Version of the software. Example : «1.0»
HISTORY_DATE	char HISTORY_DATE(N_HISTORY, N_PROF, DATE_TIME); HISTORY_DATE:long_name = "Date the history record was created"; HISTORY_DATE:conventions = "YYYYMMDDHHMISS"; HISTORY_DATE:_FillValue = " ";	Date of the action. Example : 20011217160057
HISTORY_ACTION	char HISTORY_ACTION(N_HISTORY, N_PROF, STRING4); HISTORY_ACTION:long_name = "Action performed on data"; HISTORY_ACTION:conventions = "Argo reference table 7"; HISTORY_ACTION:_FillValue = " ";	Name of the action. The action codes are described in reference table 7. Example : QCF\$ for QC failed
HISTORY_PARAMETER	char HISTORY_PARAMETER(N_HISTORY, N_PROF, STRING4); HISTORY_PARAMETER:long_name = "Station parameter action is performed on"; HISTORY_PARAMETER:conventions = " Argo reference table 3"; HISTORY_PARAMETER:_FillValue = " ";	Name of the parameter on which the action is performed. Example : PSAL
HISTORY_START_PRES	float HISTORY_START_PRES(N_HISTORY, N_PROF); HISTORY_START_PRES:long_name = "Start pressure action applied on"; HISTORY_START_PRES:_FillValue = " ";	Start pressure the action is applied to. Example : 1500.0

	99999.f; HISTORY_START_PRES:units = "decibar";	
HISTORY_STOP_PRES	float HISTORY_STOP_PRES(N_HISTORY, N_PROF); HISTORY_STOP_PRES:long_name = "Stop pressure action applied on"; HISTORY_STOP_PRES:_FillValue = 99999.f; HISTORY_STOP_PRES:units = "decibar";	Stop pressure the action is applied to. This should be greater than START_PRES. Example : 1757.0
HISTORY_PREVIOUS_VALUE	float HISTORY_PREVIOUS_VALUE(N_HISTORY, N_PROF); HISTORY_PREVIOUS_VALUE:long_name = "Parameter/Flag previous value before action"; HISTORY_PREVIOUS_VALUE:_FillValue = 99999.f;	Parameter or flag of the previous value before action. Example : 2 (probably good) for a flag that was changed to 1 (good)
HISTORY_QCTEST	char HISTORY_QCTEST(N_HISTORY, N_PROF, STRING16); HISTORY_QCTEST:long_name = "Documentation of tests performed, tests failed (in hex form)"; HISTORY_QCTEST:conventions = "Write tests performed when ACTION=QCP\$; tests failed when ACTION=QCF\$"; HISTORY_QCTEST:_FillValue = " ";	This field records the tests performed when ACTION is set to QCP\$ (qc performed), the test failed when ACTION is set to QCF\$ (qc failed). The QCTEST codes are describe in reference table 11. Example : 0A (in hexadecimal form)

Information on HISTORY_QCTEST handling

Every time a record passes through a QC test, a record is created in HISTORY. The nature of QC test is stored in HISTORY_QCTEST.

The HISTORY_ACTION is set to QCP\$ meaning QC performed.

QC performed only means that the test was performed, whether it failed or not.

If a test failed a second record is created in HISTORY. The nature of the failed test is stored in HISTORY_QCTEST.

The HISTORY_ACTION is set to QCF\$ meaning QC failed.

For example, assume the case of a profile with a test failure. The history structure would be as follows.

Note that we have used a parameter code of RCRD to indicate that the result applies to the entire record, not just to any particular parameter.

History information Example	
HISTORY_INSTITUTION	ME
HISTORY_SOFTWARE	QCA1
HISTORY_SOFTWARE_RELEASE	1.0
HISTORY_ACTION	QCP\$
HISTORY_PARAMETER	RCRD
HISTORY_START_PRES	99999.
HISTORY_STOP_PRES	99999.
HISTORY_PREVIOUS_VALUE	99999.
HISTORY_QCTEST	6A34CFE
HISTORY_INSTITUTION	ME

HISTORY_SOFTWARE	QCA1
HISTORY_SOFTWARE_RELEASE	1.0
HISTORY_ACTION	QCF\$
HISTORY_PARAMETER	RCRD
HISTORY_START_PRES	99999.
HISTORY_STOP_PRES	99999.
HISTORY_PREVIOUS_VALUE	99999.
HISTORY_QCTEST	0A

2.3. Trajectory format 2.1

An Argo trajectory file contains all received locations of an Argo float. There is one trajectory file per float.

In addition to locations, a trajectory file may contain measurements such as temperature, salinity or conductivity performed at some or all locations.

For file naming conventions, see §4.1 .

2.3.1. Dimensions and definitions

Name	Definition	Comment
DATE_TIME	DATE_TIME = 14;	This dimension is the length of an ASCII date and time value. Date_time convention is : YYYYMMDDHHMISS YYYY : year MM : month DD : day HH : hour of the day MI : minutes SS : seconds Date and time values are always in universal time coordinates (UTC). Examples : 20010105172834 : January 5 th 2001 17:28:34 19971217000000 : December 17 th 1997 00:00:00
STRING256 STRING64 STRING32 STRING16 STRING8 STRING4 STRING2	STRING256 = 256; STRING64 = 64; STRING32 = 32; STRING16 = 16; STRING8 = 8; STRING4 = 4; STRING2 = 2;	String dimensions from 2 to 256.
N_PARAM	N_PARAM = <int value> ;	Maximum number of parameters measured or calculated for a pressure sample. Examples : (pressure, temperature) : N_PARAM = 2 (pressure, temperature, salinity) : N_PARAM = 3 (pressure, temperature, conductivity, salinity) : N_PARAM = 4
N_MEASUREMENT	N_MEASUREMENT = unlimited;	This dimension is the number of recorded locations and measurements of the file.
N_CYCLE	N_CYCLE = <int value> ;	Maximum number of cycles performed by the float. This dimension depends on the data set. Example : N_CYCLE = 100
N_HISTORY	N_HISTORY = <int value> ;	Maximum number of history records for a location. This dimension depends on the data set Exemple : N_HISTORY = 10

2.3.2. General information on the trajectory file

This section contains information about the whole file.

Name	Definition	Comment
DATA_TYPE	char DATA_TYPE(STRING16); DATA_TYPE:comment = "Data type"; DATA_TYPE:_FillValue = " ";	This field contains the type of data contained in the file. The list of acceptable data types is in the reference table 1. Example : Argo trajectory
FORMAT_VERSION	char FORMAT_VERSION(STRING4); FORMAT_VERSION:comment = "File format version "; FORMAT_VERSION:_FillValue = " ";	File format version Example : «2.1»
HANDBOOK_VERSION	char HANDBOOK_VERSION(STRING4); HANDBOOK_VERSION:comment = "Data handbook version"; HANDBOOK_VERSION:_FillValue = " ";	Version number of the data handbook. This field indicates that the data contained in this file are managed according to the policy described in the Argo data management handbook. Example : «1.0»
REFERENCE_DATE_TIME	char REFERENCE_DATE_TIME(DATE_TIME); REFERENCE_DATE_TIME:comment = "Date of reference for Julian days"; REFERENCE_DATE_TIME:conventions = "YYYYMMDDHHMISS"; REFERENCE_DATE_TIME:_FillValue = " ";	Date of reference for julian days. The recommended reference date time is «19500101000000» : January 1 st 1950 00:00:00

2.3.3. General information on the float

This section contains general information on the float.

Name	Definition	Comment
PLATFORM_NUMBER	char PLATFORM_NUMBER(STRING8); PLATFORM_NUMBER:long_name = "Float unique identifier"; PLATFORM_NUMBER:conventions = "WMO float identifier : A9IIIII"; PLATFORM_NUMBER:_FillValue = " ";	WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique. Example : 6900045
PROJECT_NAME	char PROJECT_NAME(STRING64); PROJECT_NAME:comment = "Name of the project"; PROJECT_NAME:_FillValue = " ";	Name of the project which operates the float that performed the trajectory. Example : GYROSCOPE (EU project for ARGO program)
PI_NAME	char PI_NAME (STRING64); PI_NAME:comment = "Name of the principal investigator"; PI_NAME:_FillValue = " ";	Name of the principal investigator in charge of the float. Example : Yves Desaubies
TRAJECTORY_PARAMETERS	char TRAJECTORY_PARAMETERS(N_PARAM_STRING4); TRAJECTORY_PARAMETERS:long_name = "List of available parameters for the station"; TRAJECTORY_PARAMETERS:conventions = "Argo reference table 3"; TRAJECTORY_PARAMETERS:_FillValue = " ";	List of parameters contained in this trajectory file. The parameter names are listed in reference table 3. Examples : TEMP, PSAL, CNDC TEMP : temperature PSAL : practical salinity CNDC : conductivity
DATA_CENTRE	char DATA_CENTRE(STRING2); DATA_CENTRE:long_name = "Data centre in charge of float data processing"; DATA_CENTRE:conventions = "Argo reference table 4"; DATA_CENTRE:_FillValue = " ";	Code for the data centre in charge of the float data management. The data centre codes are described in the reference table 4. Example : ME for MEDS
DATE_CREATION	char DATE_CREATION(DATE_TIME); DATE_CREATION:comment = "Date of	Date and time (UTC) of creation of this file. Format : YYYYMMDDHHMISS

	file creation ; DATE_CREATION:conventions = "YYYYMMDDHHMISS"; DATE_CREATION:_FillValue = " ";	Example : 20011229161700 : December 29 th 2001 16 :17 :00
DATE_UPDATE	char DATE_UPDATE(DATE_TIME); DATE_UPDATE:long_name = "Date of update of this file"; DATE_UPDATE:conventions = "YYYYMMDDHHMISS"; DATE_UPDATE:_FillValue = " ";	Date and time (UTC) of update of this file. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 09 :05 :00
DATA_STATE_INDICATOR	char DATA_STATE_INDICATOR(STRING4); DATA_STATE_INDICATOR:long_name = "Degree of processing the data have passed through"; DATA_STATE_INDICATOR:conventions = "Argo reference table 6"; DATA_STATE_INDICATOR:_FillValue = " ";	Degree of processing the data has passed through. The data state indicator is described in the reference table 6.
INST_REFERENCE	char INST_REFERENCE(STRING64); INST_REFERENCE:long_name = "Instrument type"; INST_REFERENCE:conventions = "Brand, type, serial number"; INST_REFERENCE:_FillValue = " ";	Information about instrument : brand, type, serial number Example : APEX-SBE 259
WMO_INST_TYPE	char WMO_INST_TYPE(STRING4); WMO_INST_TYPE:long_name = "Coded instrument type"; WMO_INST_TYPE:conventions = "Argo reference table 8"; WMO_INST_TYPE:_FillValue = " ";	Instrument type from WMO code table 1770. A subset of WMO table 1770 is documented in the reference table 8. Example : 831
POSITIONING_SYSTEM	char POSITIONING_SYSTEM(STRING8); POSITIONING_SYSTEM:long_name = "Positioning system"; POSITIONING_SYSTEM:_FillValue = " ";	Name of the system used to derive the float locations from reference table 9. Examples : ARGOS

2.3.4. Locations and measurements from the float

This section contains locations for one Argo float. It may also contain measurements performed along the trajectory.

Each field in this section has a N_MEASUREMENT dimension.

N_MEASUREMENT is the number of locations (or measurement) received from the float.

When no parameter is measured along the trajectory, N_PARAM is set to 0; any field with a N_PARAM dimension is removed from the file : PARAM, PARAM_QC, PARAM_ADJUSTED, PARAM_ADJUSTED_QC, PARAM_ADJUSTED_ERROR and TRAJECTORY_PARAMETERS.

The original data received from the float and examined by real-time quality control should be placed in the <PARAM> and the QC flags set by the real-time process should be placed in the <PARAM>_QC field. The values and flags in the <PARAM> fields should never be altered.

Each parameter can be adjusted in delayed mode. In that case <PARAM> contains the original values, <PARAM>_ADJUSTED contains the adjusted values, <PARAM>_ADJUSTED_QC contains the QC flags set by the delayed-mode process, and <PARAM>_ADJUSTED_ERROR contains the adjustment uncertainties.

Name	Definition	Comment
DATA_MODE	char DATA_MODE(N_MEASUREMENT); DATA_MODE:long_name = "Delayed mode or real time data"; DATA_MODE:conventions = "R : real time; D : delayed mode";	Indicates if the profile contains real time or delayed mode data. R : real time data D : delayed mode data
DC_REFERENCE	char DC_REFERENCE(N_MEASUREMENT STRING32); DC_REFERENCE:long_name = "Location unique identifier in data centre"; DC_REFERENCE:conventions = "Data centre convention"; DC_REFERENCE:_FillValue = " ";	Unique identifier of the location in the data centre. Data centres may have different identifier schemes. DC_REFERENCE is therefore not always unique across data centres.
JULD	double JULD(N_MEASUREMENT); JULD:long_name = "Julian day (UTC) of each measurement relative to REFERENCE_DATE_TIME"; JULD:units = "days since 1950-01-01 00:00:00 UTC"; JULD:conventions = "Relative julian days with decimal part (as parts of the day)"; JULD:_FillValue = 999999.;	Julian day of the location (or measurement). The integer part represents the day, the decimal part represents the time of the measurement. Date and time are in universal time coordinates. The julian day is relative to REFERENCE_DATE_TIME. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_QC	char JULD_QC(N_MEASUREMENT); JULD_QC:long_name = "Quality on date and time"; JULD_QC:conventions = "Argo reference table 2"; JULD_QC:_FillValue = "0";	Quality flag on JULD date and time. The flag scale is described in the reference table 2. Example : 1 : the date and time seems correct.
LATITUDE	double LATITUDE(N_MEASUREMENT); LATITUDE:long_name = "Latitude of each location"; LATITUDE:units = "degree_north"; LATITUDE:_FillValue = 99999.; LATITUDE:valid_min = -90.; LATITUDE:valid_max = 90.;	Latitude of the location (or measurement). Unit : degree north Example : 44.4991 for 44° 29' 56.76" N
LONGITUDE	double LONGITUDE(N_MEASUREMENT);	Longitude of the location (or measurement).

	LONGITUDE:long_name = "Longitude of each location"; LONGITUDE:units = "degree_east"; LONGITUDE:_FillValue = 99999.; LONGITUDE:valid_min = -180.; LONGITUDE:valid_max = 180.;	Unit : degree east Example : 16.7222 for 16° 43' 19.92" E
POSITION_ACCURACY	char POSITION_ACCURACY(N_MEASUREMENT); POSITION_ACCURACY:long_name = "Estimated accuracy in latitude and longitude"; POSITION_ACCURACY:conventions = "Argo reference table 5"; POSITION_ACCURACY:_FillValue = "B";	Position accuracy received from the positioning system. The location classes from ARGOS are described in the reference table 5. Example : 3 for a latitude and longitude accuracy < 150 m.
POSITION_QC	char POSITION_QC(N_MEASUREMENT); POSITION_QC:long_name = "Quality on position"; POSITION_QC:conventions = "Argo reference table 2"; POSITION_QC:_FillValue = "0";	Quality flag on position. The flag on position is set according to (LATITUDE, LONGITUDE, JULD) quality. The flag scale is described in the reference table 2. Example : 1 : position seems correct.
CYCLE_NUMBER	int CYCLE_NUMBER(N_MEASUREMENT); CYCLE_NUMBER:long_name = "Float cycle number of the measurement"; CYCLE_NUMBER:conventions = "0..N, 0 : launch cycle, 1 : first complete cycle"; CYCLE_NUMBER:_FillValue = 99999;	Cycle number of the float for this measurement. For one cycle number, there are usually several locations/measurement received. Example : 17 for measurements performed during the 17 th cycle of the float.
<PARAM>	float <PARAM>(N_MEASUREMENT); <PARAM>:long_name = "<X>"; <PARAM>:_FillValue = <X>; <PARAM>:units = "<X>"; <PARAM>:valid_min = <X>; <PARAM>:valid_max = <X>; <PARAM>:comment = "<X>"; <PARAM>:C_format = "<X>"; <PARAM>:FORTRAN_format = "<X>"; <PARAM>:resolution = <X>;	<PARAM> contains the original values of a parameter listed in reference table 3. <X> : this field is specified in the reference table 3.
<PARAM>_QC	char <PARAM>_QC(N_MEASUREMENT); <PARAM>_QC:long_name = "quality flag"; <PARAM>_QC:conventions = "Argo reference table 2"; <PARAM>_QC:_FillValue = "0"	Quality flag applied on each <PARAM> values. The flag scale is specified in table 2.
<PARAM>_ADJUSTED	float <PARAM>_ADJUSTED(N_MEASUREMENT); <PARAM>_ADJUSTED:long_name = "<X>"; <PARAM>_ADJUSTED:_FillValue = <X>; <PARAM>_ADJUSTED:units = "<X>"; <PARAM>_ADJUSTED:valid_min = <X>; <PARAM>_ADJUSTED:valid_max = <X>; <PARAM>_ADJUSTED:comment = "<X>"; <PARAM>_ADJUSTED:C_format = "<X>"; <PARAM>_ADJUSTED:FORTRAN_format = "<X>"; <PARAM>_ADJUSTED:resolution= <X>;	<PARAM>_ADJUSTED contains the adjusted values derived from the original values of the parameter. <X> : this field is specified in the reference table 3. <PARAM>_ADJUSTED is not mandatory.
<PARAM>_ADJUSTED_QC	char <PARAM>_ADJUSTED_QC(N_MEASUREMENT); <PARAM>_ADJUSTED_QC:long_name = "quality flag"; <PARAM>_ADJUSTED_QC:conventions = "Argo reference table 2"; <PARAM>_ADJUSTED_QC:_FillValue = "0"	Quality flag applied on each <PARAM>_ADJUSTED values. The flag scale is specified in reference table 2. <PARAM>_ADJUSTED_QC is not mandatory.
<PARAM>_ADJUSTED_ERROR	float <PARAM>_ADJUSTED_ERROR(N_MEASUREM ENT); <PARAM>_ADJUSTED_ERROR:long_name = "<X>"; <PARAM>_ADJUSTED_ERROR:_FillValue = <X>; <PARAM>_ADJUSTED_ERROR:units = "<X>"; <PARAM>_ADJUSTED_ERROR:comment = "Contains the error on the adjusted values as determined by the scientific QC process. This	<PARAM>_ADJUSTED_ERROR contains the error on the adjusted values of the parameter. <X> : this field is specified in the reference table 3. <PARAM>_ADJUSTED_ERROR is not mandatory.

	field is not mandatory if scientific QC has not been applied."; <PARAM>_ADJUSTED_ERROR:C_format = "<X>"; <PARAM>_ADJUSTED_ERROR:FORTRAN_format = "<X>"; <PARAM>_ADJUSTED_ERROR:resolution=<X>;	
--	---	--

2.3.5. Cycle information from the float

This section contains information on the cycles performed by the float.

Each field in this section has a N_CYCLE dimension.

N_CYCLE is the number of cycles performed by the float.

Name	Definition	Comment
JULD_ASCENT_START	double JULD_ASCENT_START(N_CYCLE); JULD_ASCENT_START:long_name = "Start date of the ascending profile"; JULD_ASCENT_START:units = " days since 1950-01-01 00:00:00 UTC"; JULD_ASCENT_START:conventions = " Relative julian days with decimal part (as part of day) "; JULD_ASCENT_START:_FillValue=999999.f;	Julian day (UTC) of the beginning of the ascending profile. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_ASCENT_START_STATUS	char JULD_ASCENT_START_STATUS(N_CYCLE); JULD_ASCENT_START_STATUS:conventions = "0 : Nominal, 1 : Estimated, 2 :Transmitted";	0 : date comes from the float meta data 1 : date is estimated 2 : date is transmitted by the float 9 : date is unknown
JULD_ASCENT_END	double JULD_ASCENT_END(N_CYCLE); JULD_ASCENT_END:long_name = "End date of the ascending profile"; JULD_ASCENT_END:units = " days since 1950-01-01 00:00:00 UTC"; JULD_ASCENT_END:conventions = " Relative julian days with decimal part (as part of day) "; JULD_ASCENT_END:_FillValue=999999.f;	Julian day (UTC) of the end of the ascending profile. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_ASCENT_END_STATUS	char JULD_ASCENT_END_STATUS(N_CYCLE); JULD_ASCENT_END_STATUS:conventions = "0 : Nominal, 1 : Estimated, 2 :Transmitted";	0 : date comes from the float meta data 1 : date is estimated 2 : date is transmitted by the float 9 : date is unknown
JULD_DESCENT_START	double JULD_DESCENT_START(N_CYCLE); JULD_DESCENT_START:long_name = "Descent start date of the cycle"; JULD_DESCENT_START:units = " days since 1950-01-01 00:00:00 UTC"; JULD_DESCENT_START:conventions = " Relative julian days with decimal part (as part of day) "; JULD_DESCENT_START:_FillValue=999999.f;	Julian day (UTC) of the beginning of the descending profile. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_DESCENT_START_STATUS	char JULD_DESCENT_START_STATUS(N_CYCLE); JULD_DESCENT_START_STATUS:conventions = "0 : Nominal, 1 : Estimated, 2	0 : date comes from the float meta data 1 : date is estimated 2 : date is transmitted by the float 9 : date is unknown

	:Transmitted";	
JULD_DESCENT_EN D	double JULD_DESCENT_END(N_CYCLE); JULD_DESCENT_END:long_name = "Descent end date of the cycle"; JULD_DESCENT_END:units = " days since 1950-01-01 00:00:00 UTC"; JULD_DESCENT_END:conventions = " Relative julian days with decimal part (as part of day) "; JULD_DESCENT_END:_FillValue=999999. f;	Julian day (UTC) of the end of the descending profile. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_DESCENT_EN D_STATUS	char JULD_DESCENT_END_STATUS(N_CYCLE) ; JULD_DESCENT_END_STATUS:conven tions = "0 : Nominal, 1 : Estimated, 2 :Transmitted";	0 : date comes from the float meta data 1 : date is estimated 2 : date is transmitted by the float 9 : date is unknown
JULD_START_TRAN SMISSION	double JULD_START_TRANSMISSION(N_CYCLE); JULD_START_TRANSMISSION:long_name = "Start date of transmission"; JULD_START_TRANSMISSION:units = " days since 1950-01-01 00:00:00 UTC"; JULD_START_TRANSMISSION:convention s = " Relative julian days with decimal part (as part of day) "; JULD_START_TRANSMISSION:_FillValue= 999999.f;	Julian day (UTC) of the beginning of data transmission. Example : 18833.8013889885 : July 25 2001 19:14:00
JULD_START_TRAN SMISSION_STATUS	char JULD_START_TRANSMISSION_STATUS(N _CYCLE); JULD_START_TRANSMISSION_STATUS:c onventions = "0 : Nominal, 1 : Estimated, 2 :Transmitted";	0 : date comes from the float meta data 1 : date is estimated 2 : date is transmitted by the float 9 : date is unknown
GROUNDING	char GROUNDED(N_CYCLE); GROUNDED:long_name = "Did the profiler touch the ground for that cycle"; GROUNDED:conventions = "Y,N,U";	GROUNDED indicates if the float touched the ground for that cycle. Format : Y, N, U Examples : Y : yes the float touched the ground N : no U : unknown

2.3.6. History information

This section contains history information for each action performed on each measurement.

Each item of this section has a N_MEASUREMENT (number of locations or measurements), N_HISTORY (number of history records) dimension.

Name	Definition	Comment
HISTORY_INSTITUTION	char HISTORY_INSTITUTION (N_MEASUREMENT, N_HISTORY, STRING4); HISTORY_INSTITUTION:long_name = "Institution which performed action"; HISTORY_INSTITUTION:conventions = "Argo reference table 4"; HISTORY_INSTITUTION:_FillValue = " ";	Institution that performed the action. Institution codes are described in reference table4. Example : ME for MEDS

HISTORY SOFTWARE	char HISTORY SOFTWARE(N_MEASUREMENT, N_HISTORY, STRING4); HISTORY SOFTWARE:long_name = "Software which performed action"; HISTORY SOFTWARE:conventions = "Institution dependent"; HISTORY SOFTWARE:_FillValue = " ";	Name of the software that performed the action. Example : QCA1
HISTORY SOFTWARE RELEASE	char HISTORY SOFTWARE_RELEASE(N_MEASUREMENT, N_HISTORY, STRING4); HISTORY SOFTWARE_RELEASE:long_name = "Version/release of software which performed action"; HISTORY SOFTWARE_RELEASE:conventions = "Institution dependent"; HISTORY SOFTWARE_RELEASE:_FillValue = " ";	Version of the software. Example : «1.0»
HISTORY_DATE	char HISTORY_DATE(N_MEASUREMENT, N_HISTORY, DATE_TIME); HISTORY_DATE:long_name = "Date the history record was created"; HISTORY_DATE:conventions = "YYYYMMDDHHMISS"; HISTORY_DATE:_FillValue = " ";	Date and time (UTC) of the action. Example : 20011217160057
HISTORY_ACTION	char HISTORY_ACTION(N_MEASUREMENT, N_HISTORY, STRING4); HISTORY_ACTION:long_name = "Action performed on data"; HISTORY_ACTION:conventions = "Argo reference table 7"; HISTORY_ACTION:_FillValue = " ";	Name of the action. The action codes are described in reference table 7. Example : QCF\$ for quality control failed
HISTORY_PARAMETER	char HISTORY_PARAMETER(N_MEASUREMENT, N_HISTORY, STRING4); HISTORY_PARAMETER:long_name = "Station parameter action is performed on"; HISTORY_PARAMETER:conventions = "Argo reference table 3"; HISTORY_PARAMETER:_FillValue = " ";	Name of the parameter on which the action is performed. Example : PSAL
HISTORY_PREVIOUS_VALUE	float HISTORY_PREVIOUS_VALUE(N_MEASUREMENT, N_HISTORY); HISTORY_PREVIOUS_VALUE:long_name = "Parameter/Flag previous value before action"; HISTORY_PREVIOUS_VALUE:_FillValue = 99999.f;	Parameter or flag of the previous value before action. Example : 2 (probably good) for a flag that was changed to 1 (good)
HISTORY_QCTEST	char HISTORY_QCTEST(N_MEASUREMENT, N_HISTORY, STRING16); HISTORY_QCTEST:long_name = "Documentation of tests performed, tests failed (in hex form)"; HISTORY_QCTEST:conventions = "Write tests performed when ACTION=QCP\$; tests failed when ACTION=QCF\$"; HISTORY_QCTEST:_FillValue = " ";	This field records the tests performed when ACTION is set to QCP\$ (qc performed), the test failed when ACTION is set to QCF\$ (qc failed). The QCTEST codes are describe in reference table 11. Example : 0A (in hexadecimal form)

2.4. Meta-data format 2.1

An Argo meta-data file contains information about an Argo float.

For file naming conventions, see §4.1 .

2.4.1. Dimensions and definitions

Name	Definition	Comment
DATE_TIME	DATE_TIME = 14;	This dimension is the length of an ASCII date and time value. Date_time convention is : YYYYMMDDHHMISS YYYY : year MM : month DD : day HH : hour of the day MI : minutes SS : seconds Date and time values are always in universal time coordinates (UTC). Examples : 20010105172834 : January 5 th 2001 17:28:34 19971217000000 : December 17 th 1997 00:00:00
STRING256 STRING64 STRING32 STRING16 STRING8 STRING4 STRING2	STRING256 = 256; STRING64 = 64; STRING32 = 32; STRING16 = 16; STRING8 = 8; STRING4 = 4; STRING2 = 2;	String dimensions from 2 to 256.
N_CYCLES	N_CYCLES = <int value> ;	Number of different nominal cycles. This value is usually set to 1 : all the cycles are programmed to be the same. However, some floats may perform cycles with different programming. Example : a float is programmed to perform regularly 4 cycles with 400 decibar profiles and the 5 th cycle with a 2000 decibar profile. In that case, N_CYCLE is set to 2. N_CYCLES = 2 The first N_CYCLE has a REPETITION_RATE of 4 and the second has a REPETITION_RATE of 1.
N_PARAM	N_PARAM=<int value> ;	Number of parameters measured or calculated for a pressure sample. Examples : (pressure, temperature) : N_PARAM = 2 (pressure, temperature, salinity) : N_PARAM = 3 (pressure, temperature, conductivity, salinity) : N_PARAM = 4

2.4.2. General information on the meta-data file

This section contains information about the whole file.

Name	Definition	Comment
DATA_TYPE	char DATA_TYPE(STRING16); DATA_TYPE:comment = "Data type"; DATA_TYPE:_FillValue = " ";	This field contains the type of data contained in the file. The list of acceptable data types is in the reference table 1. Example : Argo meta-data
FORMAT_VERSION	char FORMAT_VERSION(STRING4); FORMAT_VERSION:comment = "File format version "; FORMAT_VERSION:_FillValue = " ";	File format version Example : «2.1»
HANDBOOK_VERSION	char HANDBOOK_VERSION(STRING4); HANDBOOK_VERSION:comment = "Data handbook version"; HANDBOOK_VERSION:_FillValue = " ";	Version number of the data handbook. This field indicates that the data contained in this file are managed according to the policy described in the Argo data management handbook. Example : «1.0»
DATE_CREATION	char DATE_CREATION(DATE_TIME); DATE_CREATION:comment = "Date of file creation "; DATE_CREATION:conventions = "YYYYMMDDHHMISS"; DATE_CREATION:_FillValue = " ";	Date and time (UTC) of creation of this file. Format : YYYYMMDDHHMISS Example : 20011229161700 : December 29 th 2001 16:17:00
DATE_UPDATE	char DATE_UPDATE(DATE_TIME); DATE_UPDATE:long_name = "Date of update of this file"; DATE_UPDATE:conventions = "YYYYMMDDHHMISS"; DATE_UPDATE:_FillValue = " ";	Date and time (UTC) of update of this file. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 09:05:00

2.4.3. Float characteristics

This section contains the main characteristics of the float.

Name	Definition	Comment
PLATFORM_NUMBER	char PLATFORM_NUMBER(STRING8); PLATFORM_NUMBER:long_name = "Float unique identifier"; PLATFORM_NUMBER:conventions = "WMO float identifier : A9IIIII"; PLATFORM_NUMBER:_FillValue = " ";	WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique. Example : 6900045
PTT	char PTT (STRING256); PTT:long_name = "Transmission identifier (ARGOS, ORBCOMM, etc.)"; PTT:_FillValue = " ";	Transmission identifier of the float. Comma separated list for multi-beacon transmission. Example : 22507 : the float is equipped with one ARGOS beacon. 22598,22768 : the float is equipped with 2 ARGOS beacons.
TRANS_SYSTEM	char TRANS_SYSTEM(STRING16); TRANS_SYSTEM:long_name = "The telecommunications system used"; TRANS_SYSTEM:_FillValue = " ";	Name of the telecommunication system from reference table 10. Example : ARGOS
TRANS_SYSTEM_ID	char TRANS_SYSTEM_ID(STRING32); TRANS_SYSTEM_ID:long_name = "The program identifier used by the transmission system"; TRANS_SYSTEM_ID:_FillValue = " ";	Program identifier of the telecommunication subscription. Example : 38511 is a program number for all the beacons of an ARGOS customer.
TRANS_FREQUENCY	char TRANS_FREQUENCY(STRING16); TRANS_FREQUENCY:long_name = "The frequency of transmission from the float"; TRANS_FREQUENCY:units = "hertz"; TRANS_FREQUENCY:_FillValue = "9";	Frequency of transmission from the float. Unit : hertz Example : ...
TRANS_REPETITION	float TRANS_REPETITION; TRANS_REPETITION:long_name = "The repetition rate of transmission from the float"; TRANS_REPETITION:units = "second"; TRANS_REPETITION:_FillValue = 99999.f;	Repetition rate of the transmission system. Unit : second Example : 40 for a repetition of messages every 40 seconds.
POSITIONING_SYSTEM	char POSITIONING_SYSTEM(STRING8); POSITIONING_SYSTEM:long_name = "Positioning system"; POSITIONING_SYSTEM:_FillValue = " ";	Position system from reference table 9. ARGOS or GPS are 2 positioning systems. Example : ARGOS
CLOCK_DRIFT	float CLOCK_DRIFT; CLOCK_DRIFT:long_name = "The rate of drift of the float clock"; CLOCK_DRIFT:units = "decisecond/day"; CLOCK_DRIFT:_FillValue = "99999.f";	Rate of drift of the float internal clock. Unit : decisecond/day Example : 1.57
PLATFORM_MODEL	char PLATFORM_MODEL (STRING16); PLATFORM_MODEL:long_name = "Model of the float "; PLATFORM_MODEL:_FillValue = " ";	Model of the float. Example : APEX-SBE
PLATFORM MAKER	char PLATFORM_MAKER (STRING256); PLATFORM_MAKER:long_name = "The name of the manufacturer"; PLATFORM_MAKER:_FillValue = " ";	Name of the manufacturer. Example : Webb research
PLATFORM_SERIAL_NO	char PLATFORM_SERIAL_NO(STRING16); PLATFORM_SERIAL_NO:long_name = "The serial number of the platform "; PLATFORM_SERIAL_NO:_FillValue = " ";	Serial number of the float. Example : APEX-SBE 259
DIRECTION	char DIRECTION; DIRECTION:long_name = "Direction of the profiles"; DIRECTION:conventions = "A: ascending profiles, B: descending and ascending profiles";	Direction of the profiles of the float. A : ascending profiles only B : descending and ascending profiles
PROJECT_NAME	char PROJECT_NAME(STRING64); PROJECT_NAME:long_name = "The program under which the float was deployed"; PROJECT_NAME:_FillValue = " ";	Name of the project which operates the profiling float that performed the profile. Example : GYROSCOPE (EU project for Argo program)

DATA_CENTRE	char DATA_CENTRE(STRING2); DATA_CENTRE:long_name = "Data centre in charge of float real-time processing"; DATA_CENTRE:conventions = "Argo reference table 4"; DATA_CENTRE:_FillValue = " ";	Code of the data centre in charge of the float data management. The data centre codes are described in the reference table 4. Example : ME for MEDS
PI_NAME	char PI_NAME (STRING64); PI_NAME:comment = "Name of the principal investigator"; PI_NAME:_FillValue = " ";	Name of the principal investigator in charge of the profiling float. Example : Yves Desaubies
ANOMALY	char ANOMALY(STRING256); ANOMALY:long_name = "Describe any anomalies or problems the float may have had."; ANOMALY:_FillValue = " ";	This field describes any anomaly or problem the float may have had. Example : "the immersion drift is not stable."

2.4.4. Float deployment and mission information

Name	Definition	Comment
LAUNCH_DATE	char LAUNCH_DATE(DATE_TIME); LAUNCH_DATE:long_name = "Date (UTC) of the deployment"; LAUNCH_DATE:conventions = "YYYYMMDDHHMISS"; LAUNCH_DATE:_FillValue = " ";	Date and time (UTC) of launch of the float. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 03:05:00
LAUNCH_LATITUDE	double LAUNCH_LATITUDE; LAUNCH_LATITUDE:long_name = "Latitude of the float when deployed"; LAUNCH_LATITUDE:units = "degrees_north"; LAUNCH_LATITUDE:_FillValue = 99999.; LAUNCH_LATITUDE:valid_min = -90.; LAUNCH_LATITUDE:valid_max = 90.;	Latitude of the launch. Unit : degree north Example : 44.4991 : 44° 29' 56.76" N
LAUNCH_LONGITUDE	double LAUNCH_LONGITUDE; LAUNCH_LONGITUDE:long_name = "Longitude of the float when deployed"; LAUNCH_LONGITUDE:units = "degrees_east"; LAUNCH_LONGITUDE:_FillValue = 99999.; LAUNCH_LONGITUDE:valid_min = -180.; LAUNCH_LONGITUDE:valid_max = 180.;	Longitude of the launch. Unit : degree east Example : 16.7222 : 16° 43' 19.92" E
LAUNCH_QC	char LAUNCH_QC; LAUNCH_QC:long_name = "Quality on launch date, time and location"; LAUNCH_QC:conventions = "Argo reference table 2"; LAUNCH_QC:_FillValue = "0";	Quality flag on launch date, time and location. The flag scale is described in the reference table 2. Example : 1 : launch location seems correct.
START_DATE	char START_DATE(DATE_TIME); START_DATE:long_name = "Date (UTC) of the first descent of the float."; START_DATE:conventions = "YYYYMMDDHHMISS";	Date and time (UTC) of the first descent of the float. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 06:05:00
START_DATE_QC	char START_DATE_QC; START_DATE_QC:long_name = "Quality on start date"; START_DATE_QC:conventions = "Argo reference table 2"; START_DATE_QC:_FillValue = "0";	Quality flag on start date. The flag scale is described in the reference table 2. Example : 1 : start date seems correct.
DEPLOY_PLATFORM	char DEPLOY_PLATFORM(STRING32); DEPLOY_PLATFORM:long_name = "Identifier of the deployment platform"; DEPLOY_PLATFORM:_FillValue = " ";	Identifier of the deployment platform. Example : L'ATALANTE
DEPLOY_MISSION	char DEPLOY_MISSION(STRING32); DEPLOY_MISSION:long_name = "Identifier of the mission used to deploy the float"; DEPLOY_MISSION:_FillValue = " ";	Identifier of the mission used to deploy the platform. Example : POMME2
DEPLOY_AVAILABLE_PRO FILE_ID	char DEPLOY_AVAILABLE_PROFILE_ID(STRING256); DEPLOY_AVAILABLE_PROFILE_ID:long_name = "Identifier of stations used to verify the first profile"; DEPLOY_AVAILABLE_PROFILE_ID:_FillValue = " ";	Identifier of CTD or XBT stations used to verify the first profile. Example : 58776, 58777
END_MISSION_DATE	char END_MISSION_DATE (DATE_TIME); END_MISSION_DATE:long_name = "Date (UTC) of the end of mission of the float"; END_MISSION_DATE:conventions = "YYYYMMDDHHMISS"; END_MISSION_DATE:_FillValue = " ";	Date (UTC) of the end of mission of the float. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 03:05:00
END_MISSION_STATUS	char END_MISSION_STATUS; END_MISSION_STATUS:long_name = "Status of the end of mission of the float"; END_MISSION_STATUS:conventions = "T:No more transmission received, R:Retrieved"; END_MISSION_STATUS:_FillValue = " ";	Status of the end of mission of the float

2.4.5. Float sensor information

This section contains information about the sensors of the profiler.

Name	Definition	Comment
SENSOR	char SENSOR(N_PARAM,STRING4); SENSOR:long_name = "List of sensors on the float "; SENSOR:conventions = "Argo reference table 3"; SENSOR:_FillValue = " ";	Parameters measured by sensors of the float. The parameter names are listed in reference table 3. Examples : TEMP, PSAL, CNDC TEMP : temperature in celsius PSAL : practical salinity in psu CNDC : conductivity in mhos/m
SENSOR MAKER	char SENSOR MAKER(N_PARAM,STRING256); SENSOR MAKER:long_name = "The name of the manufacturer "; SENSOR MAKER:_FillValue = " ";	Name of the manufacturer of the sensor. Example : SEABIRD
SENSOR_MODEL	char SENSOR_MODEL (N_PARAM,STRING256); SENSOR_MODEL:long_name = "Type of sensor "; SENSOR_MODEL:_FillValue = " ";	Model of sensor. Example : Salinity sensor
SENSOR_SERIAL_NO	char SENSOR_SERIAL_NO(N_PARAM,STRING16); SENSOR_SERIAL_NO:long_name = "The serial number of the sensor "; SENSOR_SERIAL_NO:_FillValue = " ";	Serial number of the sensor. Example : SBE211
SENSOR_UNITS	char SENSOR_UNITS(N_PARAM, STRING16); SENSOR_UNITS:long_name = "The units of accuracy and resolution of the sensor"; SENSOR_UNITS:_FillValue = " ";	Units of accuracy of the sensor. Example : psu
SENSOR_ACCURACY	float SENSOR_ACCURACY(N_PARAM); SENSOR_ACCURACY:long_name = "The accuracy of the sensor"; SENSOR_ACCURACY:_FillValue = 99999.f;	Accuracy of the sensor. Example : 0.005
SENSOR_RESOLUTION	float SENSOR_RESOLUTION(N_PARAM); SENSOR_RESOLUTION:long_name = "The resolution of the sensor"; SENSOR_RESOLUTION:_FillValue = 99999.f;	Resolution of the sensor. Example : 0.001

2.4.6. Float calibration information

This section contains information about the calibration of the profiler. The calibration described in this section is an instrumental calibration. The scientific calibration, based on a data analysis is described in the profile format.

Name	Definition	Comment
PARAMETER	char PARAMETER(N_PARAM,STRING4); PARAMETER:long_name = "List of parameters with calibration information"; PARAMETER:conventions = "Argo reference table 3"; PARAMETER:_FillValue = " ";	Parameters measured on this float. The parameter names are listed in reference table 3. Examples : TEMP, PSAL, CNDC TEMP : temperature in celsius PSAL : practical salinity in psu CNDC : conductivity in mhos/m
PREDEPLOYMENT_CALIB_EQUATION	char PREDEPLOYMENT_CALIB_EQUATION(N_PARAM,STRING256); PREDEPLOYMENT_CALIB_EQUATION:long_name = "Calibration equation for this parameter"; PREDEPLOYMENT_CALIB_EQUATION:_FillValue = " ";	Calibration equation for this parameter. Example : $T_c = a_1 * T + a_0$
PREDEPLOYMENT_CALIB	char	Calibration coefficients for this equation.

_COEFFICIENT	<pre>PREDEPLOYMENT_CALIB_COEFFICIENT(N_PARAM,STRIN G256); PREDEPLOYMENT_CALIB_COEFFICIENT:long_name = "Calibration coefficients for this equation"; PREDEPLOYMENT_CALIB_COEFFICIENT:_FillValue = " ";</pre>	Example : a1=0.99997 , a0=0.0021
PREDEPLOYMENT_CALIB_COMMENT	char PREDEPLOYMENT_CALIB_COMMENT(N_PARAM,STRING2 56); PREDEPLOYMENT_CALIB_COMMENT:long_name = "Comment applying to this parameter calibration"; PREDEPLOYMENT_CALIB_COMMENT:_FillValue = " ";	Comments applying to this parameter calibration. Example : The sensor is not stable

2.4.7. Float cycle information

This section contains information on the cycle characteristics of the float. The values included in this section are programmed or estimated. They are not measured.

Each value has a N_CYCLES dimension. Each N_CYCLE describes a cycle configuration.

Name	Definition	Comment
REPETITION_RATE	int REPETITION_RATE(N_CYCLES); REPETITION_RATE:long_name = "The number of times this cycle repeats"; REPETITION_RATE:units = "number"; REPETITION_RATE:_FillValue = 99999;	Number of times this cycle repeats. Usually, REPETITION_RATE and N_CYCLE are set to 1 : all the cycles are programmed to be the same. However, some floats may perform cycles with different programming. Example : a float is programmed to perform regularly 4 cycles with 400 decibar profiles and the 5 th cycle with a 2000 decibar profile. In that case, N_CYCLE is set to 2. The first N_CYCLE has a REPETITION_RATE of 4 and the second has a REPETITION_RATE of 1.
CYCLE_TIME	float CYCLE_TIME(N_CYCLES); CYCLE_TIME:long_name = "The total time of a cycle : descent + parking + ascent + surface"; CYCLE_TIME:units = "decimal days"; CYCLE_TIME:_FillValue = 99999.f;	Total time of a cycle. This time includes the descending time, the parking time, the ascending time and the surface time. Unit : decimal day Example : 10.0 for a ten day cycle.
PARKING_TIME	float PARKING_TIME(N_CYCLES); PARKING_TIME:long_name = "The time spent at the parking pressure"; PARKING_TIME:units = "decimal days"; PARKING_TIME:_FillValue = 99999.f;	Time spent at the parking pressure. This time does not include the descending and ascending times. Unit : decimal day Example : 9.25 for 9 days and 6 hours at parking pressure.
DESCENDING_PROFILING_TIME	float DESCENDING_PROFILING_TIME(N_CYCLES); DESCENDING_PROFILING_TIME:long_name = "The time spent sampling the descending profile"; DESCENDING_PROFILING_TIME:units = "decimal hour"; DESCENDING_PROFILING_TIME:_FillValue = 99999.f;	Time spent in descent. Unit : decimal hour Example : 8.5 for 8 hours 30 minutes of descending
ASCENDING_PROFILING_TIME	float ASCENDING_PROFILING_TIME(N_CYCLES); ASCENDING_PROFILING_TIME:long_name = "The time spent sampling the ascending profile"; ASCENDING_PROFILING_TIME:units = "decimal hour"; ASCENDING_PROFILING_TIME:_FillValue = 99999.f;	Time spent in ascent. Unit : decimal hour Example : 7.5 for 7 hours 30 minutes of descending
SURFACE_TIME	float SURFACE_TIME(N_CYCLES); SURFACE_TIME:long_name = "The time spent at the surface."; SURFACE_TIME:units = "decimal hour"; SURFACE_TIME:_FillValue = 99999.f;	Time spent on the surface (surface drift). Unit : decimal hour Example : 10 for a 10 hours surface drift.
PARKING_PRESSURE	float PARKING_PRESSURE(N_CYCLES); PARKING_PRESSURE:long_name = "The pressure of subsurface drifts"; PARKING_PRESSURE:units = "decibar"; PARKING_PRESSURE:_FillValue = 99999.f;	Pressure of the subsurface drift. Unit : decibar Example : 1500.0 for a subsurface drift at 1500.0 decibars.
DEEPEST_PRESSURE	float DEEPEST_PRESSURE(N_CYCLES); DEEPEST_PRESSURE:long_name = "The deepest pressure sampled in the ascending profile"; DEEPEST_PRESSURE:units = "decibar"; DEEPEST_PRESSURE:_FillValue = 99999.f;	Deepest pressure sampled in the ascending profile. Unit : decibar Example : 2000.0 for an ascending profile starting at 2000.0 decibar.
DEEPEST_PRESSURE_DESCENDING	float DEEPEST_PRESSURE_DESCENDING(N_CYCLES); DEEPEST_PRESSURE_DESCENDING:long_name = "The deepest pressure sampled in the descending profile"; DEEPEST_PRESSURE_DESCENDING:units = "decibar"; DEEPEST_PRESSURE_DESCENDING:_FillValue = 99999.f;	Deepest pressure sampled in the descending profile. Unit : decibar Example : 500.0 for a descending profile ending at 500.0 decibar.

2.5. Technical information format 2.1

An Argo technical file contains technical information from an Argo float. This information is registered for each cycle performed by the float.

The number and the type of technical information is different from one float model to another. To be flexible, for each cycle, the name of the parameters and their values are recorded. The name of the parameters recorded may therefore change from one model of float to another.

For file naming conventions, see §4.1 .

2.5.1. Dimensions and definitions

Name	Definition	Comment
DATE_TIME	DATE_TIME = 14;	This dimension is the length of an ASCII date and time value. Date and time values are always in universal time coordinates (UTC). Date_time convention is : YYYYMMDDHHMISS <ul style="list-style-type: none"> • YYYY : year • MM : month • DD : day • HH : hour of the day • MI : minutes • SS : seconds Examples : 20010105172834 : January 5 th 2001 17:28:34 19971217000000 : December 17 th 1997 00:00:00
STRING256 STRING64, STRING32 STRING16, STRING8 STRING4 STRING2	STRING256 = 256; STRING64 = 64; STRING32 = 32; STRING16 = 16; STRING8 = 8; STRING4 = 4; STRING2 = 2;	String dimensions from 2 to 256.
N_TECH_PARAM	N_TECH_PARAM = <int value> ;	Number of technical parameters. Example : N_TECH_PARAM=25 Twenty five different parameters are recorded for each cycle.
N_CYCLE	N_CYCLE = UNLIMITED;	Number of cycles performed by the float.

2.5.2. General information on the technical data file

This section contains information about the technical data file itself.

Name	Definition	Comment
PLATFORM_NUMBER	char PLATFORM_NUMBER(STRING8); PLATFORM_NUMBER:long_name = "Float unique identifier"; PLATFORM_NUMBER:conventions = "WMO float identifier : A9IIIII"; PLATFORM_NUMBER:_FillValue = " ";	WMO float identifier. WMO is the World Meteorological Organization. This platform number is unique. Example : 6900045
DATA_TYPE	char DATA_TYPE(STRING32); DATA_TYPE:comment = "Data type"; DATA_TYPE:_FillValue = " ";	This field contains the type of data contained in the file. The list of acceptable data types is in the reference table 3. Example : Technical data
FORMAT_VERSION	char FORMAT_VERSION(STRING4); FORMAT_VERSION:comment = "File format version "; FORMAT_VERSION:_FillValue = " ";	File format version Example : «2.1»
HANDBOOK_VERSION	char HANDBOOK_VERSION(STRING4); HANDBOOK_VERSION:comment = "Data handbook version"; HANDBOOK_VERSION:_FillValue = " ";	Version number of the data handbook. This field indicates that the data contained in this file are managed according to the policy described in the Argo data management handbook. Example : «1.0»
DATA_CENTRE	char DATA_CENTRE(STRING2); DATA_CENTRE:long_name = "Data centre in charge of float data processing"; DATA_CENTRE:conventions = "Argo reference table 4"; DATA_CENTRE:_FillValue = " ";	Code of the data centre in charge of the float data management. The data centre codes are described in the reference table 4. Example : ME for MEDS
DATE_CREATION	char DATE_CREATION(DATE_TIME); DATE_CREATION:comment = "Date of file creation "; DATE_CREATION:conventions = "YYYYMMDDHHMISS"; DATE_CREATION:_FillValue = " ";	Date and time (UTC) of creation of this file. Format : YYYYMMDDHHMISS Example : 20011229161700 : December 29 th 2001 16 :17 :00
DATA_UPDATE	char DATE_UPDATE(DATE_TIME); DATE_UPDATE:long_name = "Date of update of this file"; DATE_UPDATE:conventions = "YYYYMMDDHHMISS"; DATE_UPDATE:_FillValue = " ";	Date and time (UTC) of update of this file. Format : YYYYMMDDHHMISS Example : 20011230090500 : December 30 th 2001 09 :05 :00

2.5.3. Technical data

This section contains a set of technical data for each profile.

There are N_TECH_PARAM (eg : 25) technical parameters recorded for each cycle.

For each cycle, for each technical parameter, the name of the parameter and the value of the parameter are recorded.

The parameter name and its value are recorded as strings of 32 characters.

Name	Definition	Comment
TECHNICAL_PARAMETER_NAME	<pre>char TECHNICAL_PARAMETER_NAME(N_CYCLE, N_TECH_PARAM, STRING32) TECHNICAL_PARAMETER_NAME: long_name="Name of technical parameters for this cycle"; TECHNICAL_PARAMETER_NAME: _fillValue = " ";</pre>	<p>Name of the technical parameter. Example : "Battery voltage"</p>
TECHNICAL_PARAMETER_VALUE	<pre>char TECHNICAL_PARAMETER_VALUE(N_CYCLE, N_TECH_PARAM, STRING32) TECHNICAL_PARAMETER_VALUE: long_name="Value of technical parameters for this cycle"; TECHNICAL_PARAMETER_VALUE: _fillValue = " ";</pre>	<p>Value of the technical parameter. Example : "11.5"</p>

2.6. GDAC FTP directory file format 1.0

2.6.1. Profile directory format

The directory describes the profile files of the ftp site. The format is an autodescriptive ascii with xml tags.

The directory contains :

- A list of general informations : project name, comment, version, ftp root address
- A table with a description of each file of the GDAC ftp site. This table is a comma separated list.

Profile directory format definition
<pre><?xml version="1.0" ?> <directory_profile> The directory describes the profile files of the argo GDAC ftp site. <project> Name of the project </project> <comment> Comment on the directory file </comment> <version> Version of the format of the directory file </version> <date_update> Date of update of the directory file. YYYYMMDDHHMISS </date_update> <ftp_root number="1"> Internet ftp address of the root of the GDAC. The "number" attribute is a unique integer for each GDAC. </ftp_root> <table separator=','> The "separator" attribute is used in table_content section. <table_description> Name of the columns of <table_content> file, date, latitude, longitude, ocean, profiler_type, institution, date_update • file : path and file name on the ftp site • date : date of the profile, YYYYMMDDHHMISS • latitude, longitude : location of the profile • ocean : code of the ocean of the profile as described in reference table 13 • profiler_type : type of profiling float as described in reference table 8 • institution : institution of the profiling float described in reference table 4 • date_update : date of last update of the file, YYYYMMDDHHMISS </table_description> <table_content> Each line describes a file of the gdac ftp site. Each line contains the items listed in <table_description>. </table_content> </table></pre>

```
</directory_profile>
```

Profile directory format example

```
<?xml version="1.0" ?>
<directory_profile>
  <project>Argo</project>
  <comment>Argo gdac ftp directory</comment>
  <version>1.0</version>
  <date_update>20030105203826</date_update>
  <ftp_root number="1">
    ftp://ftp.ifremer.fr/ifremer/argo/dac
  </ftp_root>
  <ftp_root number="2">
    ftp://usgodae.usgodae.org/pub/outgoing/argo/dac
  </ftp_root>
  <table separator=';'>
    <table_description>file,date,latitude,longitude,ocean,profiler_type,institution,date_update</table_description>
    <table_content>
      aoml/13857/profiles/R13857_001.nc,199707292003,0.267,-16.032,A,0845,AO,20030214155117
      aoml/13857/profiles/R13857_002.nc,199708091921,0.072,-17.659,A,0845,AO,20030214155354
      aoml/13857/profiles/R13857_003.nc,199708201845,0.543,-19.622,A,0845,AO,20030214155619
      ...
      jma/29051/profiles/R29051_025.nc,200110250010,30.280,143.238,P,846,JA,20030212125117
      jma/29051/profiles/R29051_026.nc,200111040004,30.057,143.206,P,846,JA,20030212125117
    </table_content>
  </table>
</directory_profile>
```

2.6.2. Trajectory directory format

The directory describes the trajectory files of the ftp site. The format is an autodescriptive ascii with xml tags.

The directory contains :

- A list of general informations : project name, comment, version, ftp root address
- A table with a description of each file of the GDAC ftp site. This table is a comma separated list.

Trajectory directory format definition

```
<?xml version="1.0" ?>
<directory_trajectory>
  The directory describes the trajectory files of the argo GDAC ftp site.

  <project>
    Name of the project
  </project>

  <comment>
    Comment on the directory file
  </comment>

  <version>
    Version of the format of the directory file
  </version>

  <date_update>
    Date of update of the directory file.
    YYYYMMDDHHMISS
  </date_update>

  <ftp_root number="1">
    Internet ftp address of the root of the GDAC.
    The "number" attribute is a unique integer for each GDAC.
  </ftp_root>
```

```

</ftp_root>

<table separator=','>
The "separator" attribute is used in table_content section.

<table_description>
Name of the columns of <table_content>
file, latitude_max, latitude_min, longitude_max, longitude_min, profiler_type, institution, date_update



- file : path and file name on the ftp site
- latitude_max, latitude_min, longitude_max, longitude_min : extreme locations of the float
- profiler_type : type of profiling float as described in reference table 8
- institution : institution of the profiling float described in reference table 4
- date_update : date of last update of the file, YYYYMMDDHHMISS


</table_description>

<table_content>
Each line describes a file of the gdac ftp site. Each line contains the items listed in <table_description>.

</table_content>
</table_content>

</directory_trajectory>

```

Trajectory directory format example

```

<?xml version="1.0" ?>
<directory_trajectory>
<project>Argo</project>
<comment>Argo gdac trajectory directory</comment>
<version>1.0</version>
<date_update>20030105203826</date_update>
<ftp_root number="1">
  ftp://ftp.ifremer.fr/ifremer/argo/dac
</ftp_root>
<ftp_root number="2">
  ftp://usgodaе.usgodaе.org/pub/outgoing/argo/dac
</ftp_root>
<table separator=','>
<table_description>file, latitude_max, latitude_min, longitude_max, longitude_min, profiler_type, institution, date_update </table_description>
<table_content>
  aoml/13857/13857_traj.nc,1.25,0.267,-16.032,-18.5,0.0845,AO,20030214155117
  aoml/13857/13857_traj.nc,0.072,-17.659,A,0.0845,AO,20030214155354
  aoml/13857/13857_traj.nc,0.543,-19.622,A,0.0845,AO,20030214155619
  ...
  jma/29051/29051_traj.nc,32.280,30.280,143.238,140.238,846,JA,20030212125117
  jma/29051/29051_traj.nc,32.352,30.057,143.206,140.115,846,JA,20030212125117
</table_content>
</table>
</directory_trajectory >

```

2.6.3. Meta-data directory format

The directory describes the meta-data files of the ftp site. The format is an autodescriptive ascii with xml tags.

The directory contains :

- A list of general informations : project name, comment, version, ftp root address
- A table with a description of each file of the GDAC ftp site. This table is a comma separated list.

Metadata directory format definition

```

<?xml version="1.0" ?>
<directory_metadata>

```

The directory describes the metadata files of the argo GDAC ftp site.

```

<project>
  Name of the project
</project>

<comment>
  Comment on the directory file
</comment>

<version>
  Version of the format of the directory file
</version>

<date_update>
  Date of update of the directory file.
  YYYYMMDDHHMISS
</date_update>

<ftp_root number="1">
  Internet ftp address of the root of the GDAC.
  The "number" attribute is a unique integer for each GDAC.
</ftp_root>

<table separator=','>
The "separator" attribute is used in table_content section.

<table_description>
  Name of the columns of <table_content>
  file, profiler_type, institution, date_update
  • file : path and file name on the ftp site
  • profiler_type : type of profiling float as described in reference table 8
  • institution : institution of the profiling float described in reference table 4
  • date_update : date of last update of the file, YYYYMMDDHHMISS

</table_description>

<table_content>
  Each line describes a file of the gdac ftp site. Each line contains the items listed in <table_description>.

</table_content>
</table>

</directory_metadata>

```

Metadata directory example

```

<?xml version="1.0" ?>
<directory_metadata>
  <project>Argo</project>
  <comment>Argo gdac trajectory directory</comment>
  <version>1.0</version>
  <date_update>20030105203826</date_update>
  <ftp_root number="1">
    ftp://ftp.ifremer.fr/ifremer/argo/dac
  </ftp_root>
  <ftp_root number="2">
    ftp://usgodaes.usgdaes.org/pub/outgoing/argo/dac
  </ftp_root>
  <table separator=','>
    <table_description>file, profiler_type, institution, date_update</table_description>
    <table_content>
      aoml/13857/13857_meta.nc,0845,AO,20030214155117
      aoml/13857/13857_meta.nc,0845,AO,20030214155354
      aoml/13857/13857_meta.nc,0845,AO,20030214155619
      ...
      jma/29051/29051_meta.nc,846,JA,20030212125117
      jma/29051/29051_meta.nc,846,JA,20030212125117
    </table_content>
  </table>
</directory_metadata>

```

3. Reference tables

3.1. Reference table 1 : data type

The following table contains the list of acceptable contents for DATA_TYPE field.

Name
Argo profile
Argo trajectory
Argo meta-data
Argo technical data

3.2. Reference table 2 : quality control flag scale

Code	Meaning	Real-time comment	Delayed mode comment
0	No QC was performed		
1	Good data	All Argo real-time tests passed and a visual QC was performed.	For delayed-mode data, this means good within the supplied error bounds.
2	Probably good data	All real-time tests passed but no visual QC was performed.	Any inconsistencies or test failures are considered insignificant after scientific evaluation against local variability.
3	Bad data that are potentially correctable	Vertical profiles suggest significant jumps or drifts relative to initial instrument calibration or neighbouring profiles, or significant deviations from climatology. These data are not to be used without scientific correction.	
4	Bad data	Data have failed one or more real-time QC tests	
5	Value changed manually		
6	Not used		
7	Not used		
8	Interpolated value		
9	Missing value. Used when original, bad values are replaced by fill_values		

3.3. Reference table 3 : parameter code table

The following table describes the parameter codes used for Argo data management.

Code	Parameter long name	Comment	Unit	Valid min	Valid max	C_Format FORTRAN_ Format resolution	Fill value
CNDC	ELECTRICAL CONDUCTIVITY	In situ measurement	mhos/m	0.f	60.f	%10.4f F10.4 0.0001f	99999.f
DOXY	DISSOLVED OXYGEN	In situ measurement	mmol/m**3	0.f	650.f	%9.3f F9.3 0.001f	99999.f
PRES	SEA PRESSURE	In situ measurement, sea surface = 0	decibar	0.f	12000.f	%7.1f F7.1 0.1f	99999.f
PSAL	PRACTICAL SALINITY	In situ measurement	psu	0.f	42.f	%9.3f F9.3 0.001f	99999.f
TEMP	SEA TEMPERATURE	In situ measurement	degree_Celsius	-2.f	40.f	%9.3f F9.3 0.001f	99999.f

3.4. Reference table 4 : data centres and institutions codes

Data centres and institutions	
AO	AOML, USA
BO	BODC, United Kingdom
CI	Institute of Ocean Sciences, Canada
CS	CSIRO, Australia
GE	BSH, Germany
IF	Ifremer, France
JA	JMA, Japan
JM	Jamstec, Japan
ME	MEDS, Canada
NA	NAVO, USA
PM	PMEL, USA
RU	Russia
SI	SIO, Scripps, USA
SP	Spain
VL	Far Eastern Regional Hydrometeorological Research Institute of Vladivostock, Russia
UW	University of Washington, USA
WH	Woods Hole Oceanographic Institution, USA

3.5. Reference table 5 : location classes (ARGOS)

ARGOS location classes	
Value	Estimated accuracy in latitude and longitude

3	< 150 m
2	150 m <= accuracy < 350 m
1	350 m <= accuracy < 1000 m
0	> 1000 m
A	No estimate of location accuracy
B	No estimate of location accuracy
Z	Invalid location

3.6. Reference table 6 : data state indicators

Level	Descriptor
0	Data are the raw output from instruments, without calibration, and not necessarily converted to engineering units. These data are rarely exchanged.
1	Data have been converted to values independent of detailed instrument knowledge. Automated calibrations may have been done. Data may not have full geospatial and temporal referencing, but have sufficient information to uniquely reference the data to the point of measurement.
2	Data have complete geospatial and temporal references. Information may have been compressed (e.g. subsampled, averaged, etc.) but no assumptions of scales of variability or thermodynamic relationships have been used in the processing.
3	The data have been processed with assumptions about the scales of variability or thermodynamic relationships. The data are normally reduced to regular space, time intervals with enhanced signal to noise.

Class	Descriptor	Subclass
A	No scrutiny, value judgements or intercomparisons are performed on the data. The records are derived directly from the input with no filtering, or subsampling.	- Some reductions or subsampling has been performed, but the original record is available. + Geospatial and temporal properties are checked. Geophysical values are validated. If not validated, this is clearly indicated.
B	Data have been scrutinized and evaluated against a defined and documented set of measures. The process is often automated (i.e. has no human intervention) and the measures are published and widely available.	- Measures are completely automated, or documentation is not widely available. + The measures have been tested on independent data sets for completeness and robustness and are widely accepted.
C	Data have been scrutinized fully including intra-record and intra-dataset comparison and consistency checks. Scientists have been involved in the evaluation and brought latest knowledge to bear. The procedures are published, widely available and widely accepted.	- Procedures are not published or widely available. Procedures have not undergone full scrutiny and testing. + Data are fully quality controlled, peer reviewed and are widely accepted as valid. Documentation is complete and widely available.

Data state indicator recommended use

The following table describes the processing stage of data and the value to be assigned the data state indicator (DS Indicator). It is the concatenation of level and class described above.

Processing Stage	DS Indicator
------------------	--------------

1. Data pass through a communications system and arrive at a processing centre. The data resolution is the highest permitted by the technical constraints of the floats and communications system.	0A (note 1)
2. The national centre assembles all of the raw information into a complete profile located in space and time.	1A (note 2)
3. The national centre passes the data through automated QC procedures and prepares the data for distribution on the GTS, to global servers and to PIs.	2B
4. Real-time data are received at global data centres that apply QC including visual inspection of the data. These are then distributed to users in near real-time	2B+ (note 3)
5. Data are reviewed by PIs and returned to processing centres. The processing centres forward the data to the global Argo servers.	2C
6. Scientists accept data from various sources, combine them as they see fit with other data and generate a product. Results of the scientific analysis may be returned to regional centres or global servers. Incorporation of these results improves the quality of the data.	2C+
7. Scientists working as part of GODAE generate fields of gridded products delivered in near real-time for distribution from the global servers. Generally, these products mostly will be based on data having passed through automated QC procedures.	3B (note 4)
8. Scientists working as part of GODAE generate fields of gridded products delivered with some time delay for distribution from the global servers. Generally, these products mostly will be based on data having passed through manual or more sophisticated QC procedures than employed on the real-time data.	3C

Notes

1. We need to have a pragmatic approach to what constitutes "original" or "raw" data. Despite the fact that an instrument may be capable of high sampling rates, what is reported from the instrument defines what is considered "raw". For example, Argo floats can certainly sample at finer scales than every 10 db, but because of communications, all we see for now is data at that (or worse) vertical resolution. Therefore the data "coming from the instrument" is "raw" output at 10db resolution.
2. The conversion of the raw data stream from the communications system into profiles of variables causes the data state indicator to switch from level 0 to 1.
3. Even though the data at global data centres use manual or semi-automated QC procedures, there is often not the intercomparisons to larger data collections and fields that would qualify the data state indicator to be set to class C. This is generally only provided by scientific scrutiny of the data.
4. The transition from class 2 to 3 occurs when assumptions of scales of variability are applied. During the course of normal data processing it is common to carry out some averaging and subsampling. This is usually done to exploit oversampling by the instrument, and to ensure good measurements are achieved. These are considered to be part of the geospatial and temporal referencing process.

3.7. Reference table 7 : history action codes

Code	Meaning
CF	Change a quality flag
CR	Create record

CV	Change value
DC	Station was checked by duplicate checking software
ED	Edit a parameter value
IP	This history group operates on the complete input record
NG	No good trace
PE	Position error. Profile position has been erroneously encoded. Corrected if possible.
QC	Quality Control
QCF\$	Tests failed
QCP\$	Test performed
SV	Set a value
TE	Time error. Profile date/time has been erroneously encoded. Corrected if possible.
UP	Station passed through the update program

3.8. Reference table 8 : instrument types

The instrument type codes comes from WMO table 1770. The WMO instrument types are available on the following web site :

http://www.meds-sdmm.dfo-mpo.gc.ca/meds/Prog_Int/J-COMM/CODES/wmhtable_e.htm#ct1770

Code Figure	Instrument
831	P-Alace float
840	Provor, no conductivity
841	Provor, Seabird conductivity sensor
842	Provor, FSI conductivity sensor
845	Webb Research, no conductivity
846	Webb Research, Seabird sensor
847	Webb Research, FSI sensor
850	Solo, no conductivity
851	Solo, Seabird conductivity sensor
852	Solo, FSI conductivity sensor
855	Ninja, no conductivity sensor
856	Ninja, SBE conductivity sensor
857	Ninja, FSI conductivity sensor
858	Ninja, TSK conductivity sensor

3.9. Reference table 9 : positioning system

Code	Description
ARGOS	ARGOS positioning system
GPS	GPS positioning system

3.10. Reference table 10 : transmission system

Code	Description

ARGOS	Argos transmission system
IRIDIUM	Iridium transmission system
ORBCOMM	Orbcomm transmission system

3.11. Reference table 11 : QC Test Ids

ID	Test
2	Platform Identification
4	Impossible Date Test
8	Impossible Location Test
16	Position on Land Test
32	Impossible Speed Test
128	Global Range Test
256	Regional Global Parameter Test
512	Pressure Increasing Test
2048	Stuck Value Test
4096	Digit Rollover Test
8192	Spike Test
16384	Top and Bottom Spike Test
32768	Gradient Test
65536	Density Inversion
131072	delayed-mode statistical test
262144	Visual QC performed by PI

3.12. Reference table 12 : history software codes

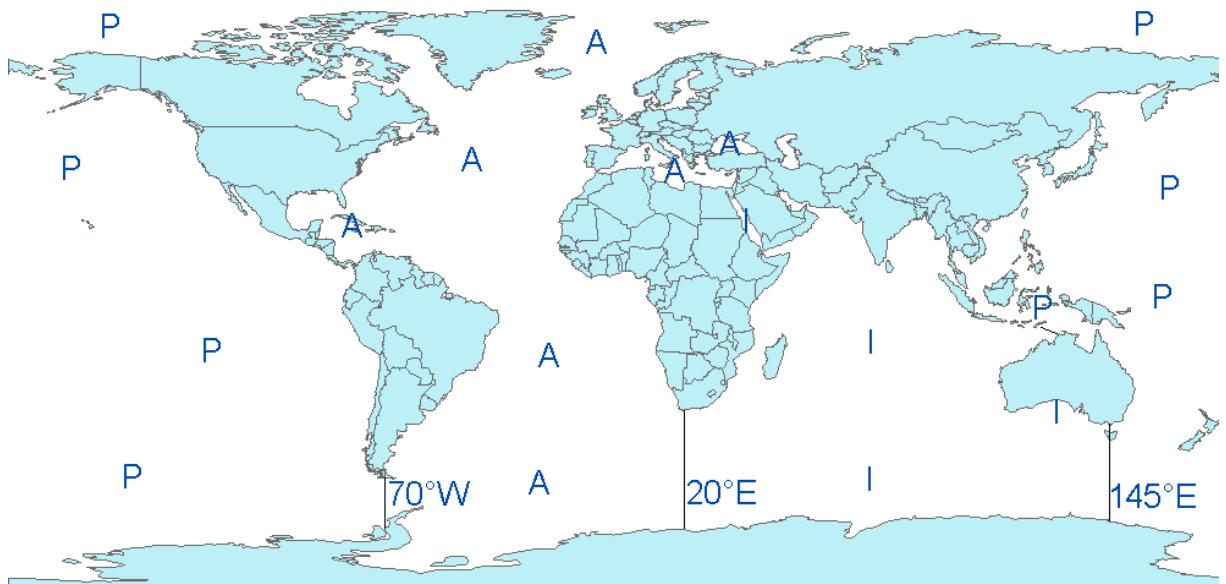
Code	Meaning
ARFM	Convert raw data from telecommunications system to a processing format
ARGQ	Automatic QC of data reported in real-time has been performed
IGO3	Checking for duplicates has been performed
ARSQ	Scientific QC has been performed
ARCA	Calibration has been performed
ARUP	Real-time data have been archived locally and sent to GDACs
ARDU	Delayed data have been archived locally and sent to GDACs

If individual centres wish to record other codes, they may add to this list as they feel is appropriate.

3.13. Reference table 13 : ocean codes

The ocean codes are used in the GDAC ftp directory files. The ocean code is not used in Argo NetCDF files.

Code	Meaning
A	Atlantic ocean area
I	Indian ocean area
P	Pacific ocean area



- The Pacific/Atlantic boundary is 70°W.
- The Pacific/Indian boundary is 145°E.
- The Atlantic/Indian boundary is 20°E.

4. Data access

The whole Argo data set is available in real time and delayed mode from the global data centres (GDACs).

The internet addresses are :

- <http://www.ifremer.fr/coriolis/cdc/argo>
- <http://www.usgodaef.nnmoc.navy.mil>

The 2 GDACs offer the same data set that is mirrored in real time.

4.1. File naming convention on GDacs

The GADC ftp sites comply with the following naming conventions :

Profile data

- <FloatID>_prof.nc for a file containing all the profiles of a float.
Example : 1900045_prof.nc
- <R/D><FloatID>_<xxx><D>.nc for an individual profile
R for Real-Time and D for Delayed-Mode data
xxx is the cycle number
If the profile is a descending profile, D is added to the cycle number
Examples : R1900045_003.nc, R1900045_003D.nc

Trajectory Data

- <FloatID>_traj.nc
Example : 1900045_traj.nc

Metadata

- <FloatID>_meta.nc
Example : 1900045_meta.nc

Technical Data

- <FloatID>_tech.nc
Example : 1900045_tech.nc

4.2. Other data sources

All Argo data are available from Argo GDACs (Global data centres).

Some Argo data are also available from GTS (Global Telecommunication System), a network operated by WMO (World Meteorological Organization).

On GTS there are 2 formats for Argo profiles :

- TESAC : an Ascii format
- BUFR : a binary format under development.

The description of these format is available from the WMO web site :

- <http://www.wmo.ch>
- <http://www.wmo.ch/web/www/DPS/NewCodesTables/WMO306vol-I-1PartA.pdf>